Towards Context-free Information Importance Estimation

Markus Zopf
Darmstadt, Germany 2019

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Informatik
Knowledge Engineering Group

Towards Context-free Information Importance Estimation

Vom Fachbereich Informatik der Technischen Universität Darmstadt genehmigte Dissertation
zur Erlangung des akademischen Grades Doctor rerum naturalium (Dr. rer. nat.)
von Markus Zopf aus Weinheim an der Bergstraße

Referenten:
Prof. Dr. Johannes Fürnkranz
Prof. Ido Dagan, Ph.D.

Tag der Einreichung: 18.12.2018
Tag der mündlichen Prüfung: 29.01.2019

URN: urn:nbn:de:tuda-tuprints-89762

Darmstadt 2019 — D17

Abstract

The amount of information contained in heterogeneous text documents such as news articles, blogs, social media posts, scientific articles, discussion forums, and microblogging platforms is already huge and is going to increase further. It is not possible for humans to cope with this flood of information, so that important information can neither be found nor be utilized. This situation is unfortunate since information is the key driver in many areas of society in the present Information Age. Hence, developing automatic means that can assist people to handle the information overload is crucial. Developing methods for automatic estimation of information importance is an essential step towards this goal.

The guiding hypothesis of this work is that prior methods for automatic information importance estimation are inherently limited because they are based on merely correlated signals that are, however, not causally linked with information importance. To resolve this issue, we lay in this work the foundations for a fundamentally new approach for importance estimation. The key idea of context-free information importance estimation is to equip machine learning models with world knowledge so that they can estimate information importance based on causal reasons.

In the first part of this work, we lay the theoretical foundations for context-free information importance estimation. First, we discuss how the abstract concept of information importance can be formally defined. So far, a formal definition of this concept is missing in the research community. We close this gap by discussing two information importance definitions, which equate the importance of information with its impact on the behavior and the impact on the course of life of the information recipients, respectively. Second, we discuss how information importance estimation abilities can be assessed. Usually, this is done by performing automatic summarization of text documents. However, we find that this approach is not ideal. Instead, we propose to consider ranking, regression, and preference prediction tasks as alternatives in future work. Third, we deduce context-free information importance estimation as a logical consequence of the previously introduced importance definitions. We find that reliable importance estimation, in particular for heterogeneous text documents, is only possible with context-free methods.

In the second part, we develop the first machine learning models based on the idea of context-free information importance estimation. To this end, we first tackle the lack of suited datasets that are required to train and test machine learning models. In particular, large and heterogeneous datasets to investigate automatic summarization of multiple source documents are missing, because their construction is complicated and costly. To solve this problem, we present a simple and cost-efficient corpus construction approach and demonstrate its applicability by creating new multi-document summarization datasets. Second, we develop a new machine learning approach for context-free information importance estimation, implement a concrete realization, and demonstrate its advantages over contextual importance estimators. Third, we develop a new method to evaluate automatic summarization methods. Previous works are based on expensive reference summaries and unreliable semantic comparisons of text documents. On the contrary, our approach uses cheap pairwise preference annotations and only much simpler sentence-level similarity estimation.

This work lays the foundations for context-free information importance estimation. We hope that future research will explore if this fundamentally new type of information importance estimation can eventually lead to human-level information importance estimation abilities.

Zusammenfassung

Die Menge an Information in heterogenen Texten wie Nachrichtenartikeln, Blogs, Beiträgen in sozialen Medien, wissenschaftlichen Artikeln, Diskussionsforen und Plattformen für Mikroblogging ist bereits heute gewaltig und wird in Zukunft weiter wachsen. Es ist für Menschen nicht möglich diese Flut von Informationen zu handhaben, sodass wichtige Informationen nicht gefunden und dadurch nicht nutzbar gemacht machen können. Dieser Umstand ist bedauerlich, da Informationen im heutigen Informationszeitalter die treibende Kraft in vielen Bereichen der Gesellschaft sind. Daher ist die Entwicklung automatischer Systeme erforderlich, die Menschen dabei unterstützen können der Informationsflut zu begegnen. Hierfür ist die Entwicklung von Methoden zur automatisierten Bewertung von Informationswichtigkeit ein wesentlicher Schritt.

Die grundlegende Hypothese in dieser Arbeit ist, dass bisherige Methoden zur automatisierten Bewertung von Informationswichtigkeit inhärent limitiert sind, da diese auf lediglich korrelierten Signalen basieren, die allerdings in keinem kausalen Zusammenhang zur Informationswichtigkeit stehen. Um dieses Problem zu lösen, werden in dieser Arbeit die Grundlagen für einen fundamental neuen Ansatz zur automatisierten Wichtigkeitsbewertung gelegt. Die Kernidee von kontextfreie Informationswichtigkeitsbewertung ist es, Modelle des maschinellen Lernens mit Weltwissen auszustatten, sodass diese auf Basis der ursächlichen Gründe die Wichtigkeit von Information bewerten können.

Im ersten Teil dieser Arbeit legen wir die theoretischen Grundlagen für kontextfreie Informationswichtigkeitsbewertung. Als erstes wird besprochen, wie der abstrakte Begriff Informationswichtigkeit formal definiert werden kann, da in der Forschungsgemeinde bisher eine klare Definition dieses Begriffes fehlt. Wir schließen diese Lücke, indem wir zwei Definitionen diskutieren, die Informationswichtigkeit mit der Auswirkung auf das Verhalten und auf das Leben der Informationsempfänger gleichsetzen. Als zweites wird diskutiert, wie die Fähigkeit zur Einschätzung der Wichtigkeit von Informationen bewerten werden kann. Üblicherweise wird dieses Problem im Kontext des automatisierten Zusammenfassens von Textdokumenten bewertet. Es zeigt sich allerdings, dass dies nicht ideal ist. Stattdessen schlagen wir vor, in zukünftiger Forschung die Erstellung von Ranglisten, die Durchführung von Regressionsanalysen und die Vorhersage von paarweise Präferenzen als Alternativen zu nutzen. Als drittes wird die kontextfreie Informationswichtigkeitsbewertung als logische Konsequenz der zuvor eingeführten Wichtigkeitsdefinitionen geschlussfolgert. Es zeigt sich, dass eine verlässliche Bewertung der Informationswichtigkeit, insbesondere in heterogenen Texten, nur mit kontextfreien Methoden möglich ist.

Im zweiten Teil entwickeln wir erste Modelle auf Basis des maschinellen Lernens zur kontextfreien Informationswichtigkeitsbewertung. Zunächst befassen wir uns hierzu mit dem Mangel an geeigneten Datensätzen, die für das Trainieren und Testen der Modelle benötigt werden. Insbesondere große und heterogene Datensätze, die nötig sind, um das automatisierte Zusammenfassen mehrerer Quelldokumente zu untersuchen, fehlen bisher, da deren Erstellung kompliziert und kostenintensiv ist. Wir lösen dieses Problem, indem wir einen einfachen und kosteneffizienten Ansatz entwickeln und seine Anwendbarkeit durch die Erstellung neuer Datensätze demonstrieren. Als zweites entwickeln wir einen neuen Ansatz des maschinellen Lernens für die kontextfreie Informationswichtigkeitsbewertung, implementieren eine konkrete Realisierung und demonstrieren die Vorteile gegenüber kontextabhängigen Systemen. Als drittes stellen wir eine neue Methode zur Evaluierung automatisierter Systeme vor. Frühere Arbeiten basieren auf teuren Referenzzusammenfassungen und unzuverlässigen semantischen Vergleichen von Textdokumenten. Unser Ansatz hingegen nutzt günstige paarweise Präferenzannotationen und einfachere semantische Vergleiche auf Satzebene.

Diese Arbeit legt den Grundstein für die kontextfreie Informationswichtigkeitsbewertung. Wir hoffen dass zukünftige Forschung erkunden wird, ob diese fundamental neue Art der Informationswichtigkeitsbewertung zu menschenähnlichen Fähigkeiten in diesem Bereich führen kann.

Contents

1 Introduction

Suppose you are a journalist, it is 2016, and your boss told you to report about the United States presidential election. You know that your target audience already has some general background knowledge about the election, such as what the election is about and who the main candidates are. Your journalistic inquiries reveal three new pieces of information, which are not yet known to the public. You discovered that

1. the U.S. Congress will certify the results on January 6, 2017
2. Donald Trump won the election and will become the 45th president
3. there have been rumors about Russian interferences in the elections

Unfortunately, the draft for the next day's newspaper is already almost full, and you cannot report all three information pieces to your audience. Instead, there is only space left for exactly one information piece. You decide to report about the outcome of the election and tell your audience that Donald Trump won and that he will become the 45th president.

Why?

As a journalist, your job is to report the most important information to your audience. You know that the outcome of the election is of significant importance to many people since the U.S. president has a lot of power. Hence, the president has a substantial impact on the life of many people in the United States and abroad which explains why many people are interested in the result of the election. The rumors about Russian interference are also unquestionable relevant. However, compared to reporting about the winner of the election, reporting about the rumors seems to be of minor importance. Finally, you know that the certification of the results in January is merely a side note and not very important to many people.

A fundamental problem for you as a journalist is to estimate the importance of newly discovered information pieces. Journalists, and humans in general, are able to estimate information importance because we have obtained a lot of general knowledge of the world and a good understanding of how the world works. In the example above, we are able to estimate the importance of reporting the winner of the election because we understand how large the consequences of the election result are. Researchers developing automatic importance estimators have largely neglected the fact that reliable importance estimation is only possible with a solid understanding of the world. Instead, the research has focused on developing automatic importance estimators that rely on unreliable heuristics. In this thesis, we lay the foundation for developing robust automatic importance estimators that are more similar to humans.

1.1 Problem Relevance

Developing machines that can estimate the importance of information automatically is of high practical relevance due to two reasons. Information is of crucial importance in the Information Age in which we are currently live in. Being well informed is essential for rational decision making in economics, politics, science, healthcare, and daily life in general. However, not all information available is important. In fact, we are drowning in a flood of information, most of which of minor importance. Due to this information overload, humans cannot identify and make use of the most important information. Hence, we need automatic means to assist humans at finding the most important information such that we can make the best use of it.

One potential application of an automatic information importance estimator is to keep users posted about an ongoing event. For example, it can be vitally important to stay up-to-date in the case of natural disasters for all an affected person. Currently, people have to check the news or various social media to get timely updates. An automatic information importance estimator would be able to scan all information sources and push a report to the users as soon as there is new important information available. Another applications could be a new type of web search engines. Currently, web search engines are used to find documents that contain information which might be helpful to satisfy a user's information need. An automatic information importance estimator could be used to render the user's manual investigation of the retrieved documents unnecessary by directly providing a short summary of the most important information contained in the documents. Such a machine would revolutionize the usability of the Internet.

Unfortunately, developing automatic information importance estimators is a difficult problem. Researchers in the area of automatic summarization have already been working on this problem issue for many decades. Reliable importance estimators are, however, still not available. As mentioned before, this thesis explores a fundamentally new type of importance estimators. We call this new type of importance estimators *context-free* since they use prior knowledge about the world to estimate information importance instead of relying on simple heuristics based on the context in which information appears in. In this thesis, we discuss six research questions which motivate context-free importance estimation and demonstrate how we can use machine learning to build context-free importance estimators.

1.2 Research Questions, Thesis Organization, and Key Contributions

The remainder of this thesis is organized into three main parts. Part I motivates context-free information importance. Part II discusses how we can use machine learning to develop context-free information importance estimators. Part III summarizes and concludes the thesis. Furthermore, implications and future research are discussed. In addition to the three main parts of this thesis, we provide detailed information about the research data management in Appendix A. In the following, we state each research question, motivate their relevance, give a brief overview of each corresponding chapter, and summarize the key contributions.

1.2.1 Part I: On Information Importance

In the first part of this thesis, we motivate the development of context-free information importance estimators by discussing the first three research questions. In Chapter 2, we discuss a definition of information importance, which is deeply rooted in the philosophy of information. The definition equates information importance with its impact on the audience's life. In Chapter 3, we discuss how information importance estimation abilities can be assessed. In particular, we argue that only using automatic summarization, which is usually considered for assessing importance estimation abilities, is suboptimal. Instead, we propose to evaluate information importance estimation based on regression, rankings, and preference prediction tasks. In Chapter 4, we motivate a fundamentally new type of information importance estimator, which does not rely on analyzing the context in which a piece of information appears. Instead, so called *context-free* importance estimation uses prior knowledge to estimate information importance. The analysis of the limitations of contextual information importance estimators motivates the development of automatic context-free information importance estimators in Part II of this thesis.

Chapter 2: How can information importance be defined in a meaningful way?

In Chapter 2, we discuss the first research question which is concerned with a meaningful definition of information importance. To this end, we start with a short introduction of the communication model in Section 2.1 that is often used to illustrate the flow of information from a sender to a receiver through a channel. Next, we review various definitions of the concept of *information* that can be found in the literature in Section 2.2. The discussed definitions have in common that they use a rather technical meaning of the concept information. Information in this technical sense does not necessarily contain any semantic information which can be understood and used by a human reader. Since we aim in this thesis for automatic means which are able to assist humans in coping with the information overload, this interpretation of the term 'information' is inappropriate in our context. Therefore, we discuss prior work on *semantic* information in Section 2.3. Semantic information is defined in this section as *meaningful data*. In addition to the qualitative definition, we also discuss a quantitative definition that equates the amount of semantic information contained in data with the amount of uncertainty reduction with respect to the state of a system. However, this information definition is also not appropriate to measure the importance of information as we demonstrate. We discuss two importance definitions in Section 2.4 which are more appropriate. The key idea of the definitions is to equate the importance of information with its impact on the behavior and the course of life of the information receiver. We also discuss the logical implications of the provided definitions. Perhaps the most important implication of this thesis is the fact that the importance of information depends mainly on the receiver of the information and not on information which surrounds the information nugget at hand, for example, in a text document or a document collection. This approach is contrary to the methods which are usually used in automatic summarization to identify important information. Section 2.5 summarizes the chapter.

Chapter 3: How can information importance estimation abilites be assessed?

In Chapter 3, we discuss how information importance estimation abilities can be assessed properly. We start by discussing the most prominent task that is related to information importance estimation, namely automatic summarization, in Section 3.1. Equipped with a precise definition 'information importance', we are able to provide a precise definition of optimal personal and optimal group summaries in Section 3.2, which was not possible before. In Section 3.4, we analyze different shortcomings of using automatic summarization as a task to estimate the information importance estimation abilities of systems and propose new tasks to test said abilities. Section 3.5 presents three new tasks based on regression, rankings, and preference prediction that can be used to assess information importance estimation abilities. The alternatives have not been used to evaluate the importance estimation abilities of summarizers so far. We summarize the chapter in Section 3.6. A key insight of this chapter is the fact that the main or major points of a document and the most important information in a document are not necessarily identical. In fact, both can diverge in heterogeneous documents to an arbitrarily large extent. The summarization community usually uses the terms main/major points and most important information synonymously due to the strong focus on newswire documents. Furthermore, we find that automatic summarization is not the only and not a necessarily good way to evaluate information importance estimation abilities. The newly presented evaluation measures may be better suited to guide research in this area in the future.

Chapter 4: Why do we need context-free information importance estimation?

In the previous chapters, we clarified how information importance can be defined and how information importance estimation abilities can be assessed. Gained insights from both previous chapters can guide the development of new information importance estimator. In Chapter 4, we focus on a specific kind of information importance estimators that estimate information importance without analyzing the context in which a particular piece of information appears in. To this end, we first clarify the meaning of the term *context* in Section 4.1. When we use the term context, we always mean the text and information which surround the information nugget whose importance has to be estimated. In extractive text summarization, for example, this can be the text before and after a sentence. Next, we introduce the terms *contextual* and *context-free* information importance estimation in the same section and discuss inherent limitations of contextual information importance estimators. We explain why currently developed summarization methods work well in newswire summarization. We conclude that this is the case because journalists introduce these features to newswire articles, and the fact that most work in automatic summarization has been conducted on newswire summarization. We contribute a simple example that requires context-free importance estimation abilities. Motivating context-free information importance estimators is the key contribution of this chapter. The key idea of context-free importance estimation is to not rely on importance signals derived from the document in which an information nugget appears in but to use previously learned domain knowledge instead. In Section 4.2, we review prior contextual importance estimators in detail. Section 4.3 summarizes this chapter.

1.2.2 Part II: Machine Learning for Information Importance Estimation

The second part of this thesis is concerned with the development of automatic context-free information importance estimators. Since it is not feasible to write algorithms that perform such a complex task directly, we develop machine learning models that are able to learn this challenging task. Every machine learning model requires three essential parts: datasets, learning algorithms, and evaluation methods. We contribute to each of the parts in Chapter 5, Chapter 6, and Chapter 7, respectively. For a short introduction to the three essential parts of machine learning, the foundations sections Section 5.1 (data), Section 6.1 (learning), and Section 7.1 (evaluation) can be read one after the other.

Chapter 5: How can challenging datasets for information importance estimation be created cheaply?

Chapter 5 focuses on the generation of new datasets that can be used to train and evaluate summarization systems. We focus in particular on large, heterogeneous, non-newswire datasets, which are difficult to solve with contextual importance estimators. In Section 5.1, we discuss relevant basic terminology. In Section 5.2, we review previously created datasets in detail and discuss their limitations. One fundamental limitation is the fact that creating multi-document summarization datasets requires a lot of human effort, which leads to small datasets. This is a serious limitation for the development of new machine learning models, which require substantial amounts of training data. The key contribution of this chapter is presented in Section 5.3. We present a new semi-automatic corpus construction approach that allows much cheaper dataset construction. The key idea of the approach is to reverse the traditional approach in which summaries are manually written for a set of source documents and start with an already available summary and search for appropriate source documents. In Section 5.4, we demonstrate the applicability of the approach where we describe the creation of the *h*MDS corpus. In Section 5.5, we develop the presented approach further to reduce the required human effort. The resulting corpus construction approach requires only minimal human effort. Consequently, it is suited to generate huge summarization datasets. We demonstrate the applicability of the modified approach and create auto-*h*MDS. Both corpora contain English and German data and have varying source document lengths and summary lengths, which are rare properties in already available corpora. We close the chapter with a short summary in Section 5.6.

Chapter 6: How can machines learn context-free estimation of information importance?

In Chapter 6, we discuss how machine learning models can be trained such that they are able to estimate information importance without considering the context in which information appears similar to the example in the introduction of this thesis. Relevant foundations are discussed in Section 6.1. In Section 6.2, we discuss prior works which are related to context-free information importance estimation. In Section 6.3, we present a concrete instantiation of context-free importance estimation. The presented model is trained such that it is able to estimate the importance of individual sentences without considering the context in which the sentences appear. Before, researchers considered mainly contextual information importance estimators. We hope that this idea will spark further exploration of context-free information importance estimators. We also introduce the idea of *contextual* pairwise preferences on which the presented model is based on. Contextual pairwise preferences extend standard pairwise preferences of the form $a \succ b$ by a context c such that we can express that a may only be preferred over b given a context c. We write $a \succ b|c$ in this case. In Section 6.4, we resolve a misconception in a prominent branch of automatic summarization called sentence regression regarding the regressand which should be used as target score. Sentence regression is due to its simplicity well suited for future work on context-free information importance estimators. The summary in Section 6.5 concludes the chapter.

Chapter 7: How can information importance estimators be evaluated automatically?

Chapter 7 is concerned with the automatic evaluation of summarization models. Evaluation is required to assess how well machine learning models perform and is a hard, unsolved task in automatic summarization similarly to the generation of summaries. In Section 7.1, we start again by discussing relevant terminology. In Section 7.2, we review prior attempts for manual and automatic evaluation. Notably, we observe a close connection of one practical evaluation method to the previously provided theoretically defined information importance. Section 7.3 reveals issues in prior assessments of the quality of evaluation methods. We show that a previously used evaluation method for evaluation systems is not reliably and may lead to wrong conclusions by providing a simple counter-example with extreme good correlation but poor performance. In Section 7.4, we present a new evaluation method which does not require reference summaries to evaluate the quality of summarization models. The key idea is to use human or automatic pairwise preferences of sentences for evaluation. In Section 7.5, we summarize the this chapter.

1.2.3 Part III: Wrap-Up

The third part concludes the thesis, provides an outlook for potential future work, and discusses limitations of the presented ideas.

In Chapter 8, we summarize the gained insights and drawn conclusions of this thesis. To this end, we revisit the six stated research questions. We also discuss opportunities for future works. In Chapter 9, we discuss some limitations of the presented idea. We conclude this thesis with some final remarks in Chapter 10.

1.3 Notation

In this section, we briefly discuss the notation used in this thesis.

Data

Tasks can be formalized as pairs of input and corresponding output. We follow the standard notion and denote the input by letter 'x' and the corresponding output by letter 'y'.

Input and output pairs appear in different levels of granularity. We denote the full manifestation of a task (i.e., the set of all possible input-output pairs) by $\mathscr{T} = (\mathscr{X}, \mathscr{Y})$. An illustration of tasks can be found in Figure 5.1 in Chapter 5.

Datasets contain usually only a subset of all pairs. We denote datasets by $\mathbf{D} = (\mathbf{X}, \mathbf{Y})$. We denote individual pairs of input and corresponding output by adding a subscript. Hence, $\mathbf{X}_i$ refers to the i-th input datapoint in the dataset and $\mathbf{Y}_i$ refers to the i-th output datapoint. An illustration of datasets can be found in Figure 5.2 in Chapter 5.

In rare cases, we talk about individual documents contained in $\mathbf{X}_i$ or $\mathbf{Y}_i$. We use the two indices in this case to denote individual documents. Hence, $\mathbf{X}_{i,j}$ refers to the j-th document in the i-th topic. For simplicity, we omit the second index if there is only one document in a topic input/output, i.e., we write $\mathbf{X}_i$ instead of $\mathbf{X}_{i,1}$.

Most of the datasets used in this thesis contain natural language text. Hence, $\mathbf{X}_i$ and $\mathbf{Y}_i$ are often text documents or sets of text documents. Text documents can often be segmented into individual sentences. We refer to individual sentences in the input and output by $\mathbf{x}_i$ and $\mathbf{y}_i$, respectively. To refer to sentences in general, which do not necessarily belong to input or output, we use letter 's'. Hence, $\mathbf{s}_i$ denotes the i-th sentence in a list of sentences. Similarly, we use letter 'w' to refer to individual words or n-grams by w_i.

Functions

We use functions in this thesis to model task. We denote functions which represent the task (i.e., the true relation from inputs to outputs) by f. Approximations of f, e.g., functions learned with machine learning, are denoted by $\tilde{f}$.

Utilities

Utilities are used in this thesis to describe the fitness of parts of the data. We use in thesis $\bar{u}$ to denote utilities of texts, $\dot{u}$ to denote utilities of parts of texts (usually sentences), and u to denote utilities of parts of sentences (e.g., words or n-grams).

Probabilities

We denote the sample space of random events, which contains all possible outcomes of a random event by Ω. Elements in Ω are denoted by ω_i. We denote random variables by X and use $\Pr(\omega_i)$ to denote the probability that X takes the outcome ω_i.

Sets

We denote sets by $\{\ldots\}$ and the empty set by $\emptyset$. Let S be a set. We use $S \cup e$ as shorthand notation for $S \cup \{e\}$ if we want to add one single element to S. Similarly, we use $S \cap e$ and $S \setminus e$ as further shorthand notations.

Furthermore, $\mathscr{P}(X) = \{U | U \subseteq X\}$ denotes the powerset of a set, i.e. the set of all subsets of a set.

Media and Information

We denote media such as texts or images by $\mathfrak{m}$. Information pieces are denoted by $\mathfrak{i}$ and sets of information nuggets are denoted by $\mathfrak{I}$. The function $\mathfrak{c}$ maps from a medium or a set of media to its/their contained information pieces. The function $\text{Imp} : \mathfrak{I} \rightarrow \mathbb{R}^+$ maps from an information nugget to its importance.

1.4 Out of Scope

The main aim of this thesis is to develop automatic methods which can assist people to find the most important information in a vast amount of information, which is not manageable manually. This aim is very challenging and this thesis can only contribute a small step towards the overall goal. There are in particular two big related problems, which are beyond the scope of this thesis. First, we do not address the problem of assessing the truthfulness of information. All information, e.g., in source documents for text summarization, is assumed to be correct and estimating a high importance for incorrect information is not considered problematic in this thesis. Obviously, only reliable information is good information and fake news are a serious problem nowadays. Hence, a good information importance estimator has also be able to distinguish correct and incorrect information. Second, desired text properties such as grammaticality, referential clarity, focus, structure, and coherence are also not in the main focus of this thesis and are only incidentally considered. Furthermore, we assume that all information that appears in text documents can be understood by the reader.

Part I

On Information Importance

In Chapter 1, we discussed that automatic information importance estimation is an important but challenging problem. Many people have a vague natural intuition of what information importance means. However, a precise definition is missing. Consequently, we discuss a definition of ’information importance’ to precisely explain what we mean by this term in Chapter 2. Based on the definition, we are able to discuss more precisely what the aim of this thesis is and how it is different from related research research.

In the second chapter in this first part, Chapter 3, we discuss how we can assess information importance estimation abilities. We focus on automatic summarization as one important application of information importance estimation. Summarization is, however, a rather suboptimal way to test how well systems are able to estimate the importance of information. Therefore, we propose alternative tasks that can guide future research.

In Chapter 4, we motive the development of *context-free* information importance estimators. To this end, we first explain what ’context-free’ means in the context of this thesis. The key idea of context-free information importance is to estimate the importance of information based on the information itself instead of estimating its importance with signals derived from the document (i.e., context) in which it appears in. We then provide a simple example to demonstrate the limitations of summarization systems that use contextual importance signals (i.e., signals which are computed by considering the document in which an information nugget appears in).

Equipped with a definition of what information importance means and a motivation for context-free information importance estimators, we continue in Part II with the development of automatic context-free information importance estimation with machine learning.

2 A Formal Theory of Information Importance

Fundamental for proper scientific discussions is a clear and precise definition of key terms. Imprecise terminology leads to disagreement based on misunderstanding. Imprecisely defined terms are a particularly serious issue for ubiquitous terms for which people have an intuitive idea but no explicitly expressed definition is available. This chapter aims at preventing misunderstandings and thus allowing a fruitful discussion of context-free information importance estimation by answering the first research question of this thesis: **How can information importance be defined in a meaningful way?**

In Section 2.1, we start by briefly reviewing the well-known communication setup with which the transfer of information can be illustrated.

Before we can talk precisely about information importance, we should have a good understanding of what we mean by ’information’. To clarify this term, we first review three well-known definitions of information in Section 2.2, namely Shannon information in Section 2.2.1, Fisher information in Section 2.2.2, and Kolmogorov information in Section 2.2.3. We analyze in Section 2.2.4 the limitations of the reviewed information definitions.

Section 2.3 addresses the limitations of the previously reviewed information definitions by discussing semantic information, which is inherently different from syntactic information. In this section, we adopt the definition by Floridi (2010) and consider information as meaningful data. In Section 2.3.1, We review how data can be defined qualitatively and discuss how the amount of data contained in a system can be measured in Section 2.3.2. In Section 2.3.3, we discuss how the amount of semantic information can be estimated. In Section 2.3.4, we analyze that neither the amount of data nor the amount of semantic information is a good indicator of the importance of information.

Next, and most important in this chapter, we discuss information importance in Section 2.4. We start by introducing an example that will be used in the section to illustrate the effects of the proposed information importance definitions. We then provide two definitions for information importance in Section 2.4.1 and Section 2.4.2. The key idea of the definitions is to equate information importance with its impact on the behavior or the course of life of the recipients of the information, respectively. We discuss logical implications of one information importance definition in Section 2.4.3 and demonstrate their effects with examples in Section 2.4.4.

Finally, we close with a short summary of the chapter in Section 2.5.

2.1 Foundations of Communication

The term ’information’ can be illustrated in the context of a communication (Shannon, 1948). A communication system contains three essential elements:

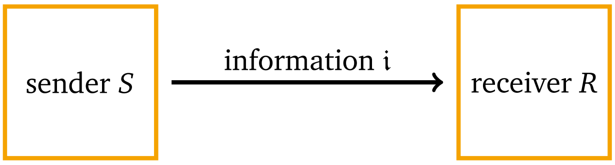


Figure 2.1.: Illustration of the communication setup. A sender S sends information i to a receiver S.

1. a *sender* S that wants to transfer information,
2. a *receiver* R that wants to receive the information, and
3. the *information* i that is transfered.

The sender has the intention to transfer information. The sender does not have to be a person but can be any system that is able to store or to produce information. A random number generator can, for example, also be considered as a source of information. The information is transmitted over a channel (which is not crucially relevant to this theses and therefore not further discussed) and received by a receiver of the information.

Problems with respect to the correctness of transmission can be observed at different levels as already argued by Weaver (1949) and Weinberger (2002): technical, semantic, and pragmatic levels. The technical level is concerned with the transmission of data. The semantic level is concerned with the transmission of meaning. Last but not least, the pragmatic level is concerned with the change of conduct of the receiver. In particular, the third level emphasizes that „the purpose of all communication is to influence the conduct of the receiver“ (Weaver, 1949). Hence, the third level is concerned with the question of whether the desired behavioral change has successfully been achieved by the communication. Figure 2.1 illustrates the communication setup.

The quantitative approaches, which are discussed in the following, fit to the sender-transmission-receiver model in different ways. They all have in common that they are concerned with measuring the amount of information. Formally, let i be a message, transmission, or information piece in the communication system and let $\mathrm{Inf}(i)$ be a function that assigns a value to i. We interpret $\mathrm{Inf}(i)$ as a function that measures the amount of information contained in i. In the following, we discuss different ways how to instantiate $\mathrm{Inf}(i)$.

2.2 Syntactic Information

In this section, we review three well-known information definitions which are concerned with the first, technical level of communication. As stated above, they provide different instantiations of the function $\mathrm{Inf}(i)$.

2.2.1 Shannon Information

One of the best-known definitions of information is based on Claude Shannon's mathematical theory of communication (MTC) (Shannon, 1948; Shannon & Weaver, 1964). We call this first information definition *Shannon information* and denote it by $\mathrm{Inf}_{\mathrm{Sh}}$. The key idea of Shannon information is to equate the amount of information contained in a string with the probability of observing the string. Intuitively speaking, this means that rare events carry much information, and frequent events carry only a little information.

Shannon information defined the amount of information carried by outcomes of random events ω_i. Hence, $\imath$ refers to outcomes of random events in this definition of information. Formally, let $X : \Omega \rightarrow \mathbb{R}$ be a random variable and let $\omega_i \in \Omega$ events which occur with probability $\Pr(\omega_i)$.

Definition 1 (Shannon information). The *Shannon information* of event $\omega_i \in \Omega$ is defined as

$$\mathrm{Inf}_{\mathrm{Sh}}(\omega_i) = \log\left(\frac{1}{\Pr(\omega_i)}\right) = -\log(\Pr(\omega_i)). \tag{2.1}$$

The average Shannon information (also called average surprise (Shannon, 1948)) created by a random process per outcome equals to Shannon's entropy, which is defined below.

Definition 2 (Shannon entropy). The Shannon entropy of a random variable X with n possible outcomes ω_i is defined as

$$H(X) = -\sum_{i=1}^{n} \Pr(\omega_i) \cdot \log(\Pr(\omega_i)). \tag{2.2}$$

Shannon chose the name entropy inspired by the Gibbs entropy (Gibbs, 1906) which is used in thermodynamics to describe how much a system is disordered. A system with a high entropy produces events that are more informative on average than a system with low entropy. It can easily be shown that the highest entropy is obtained if all symbols produced by a system are equally likely.

Consider Bernoulli trials as an example of Shannon information. A Bernoulli trial is a random experiment with exactly two possible outcomes. Let the outcomes be $\Omega = \{0, 1\}$. The Bernoulli trial is parametrized by the parameter $\theta \in [0, 1]$, which specifies the probability of observing the outcome 0. Hence, $\Pr(X = 0) = \theta$ which implies that $\Pr(X = 1) = 1 - \theta$ since $\sum_{\Omega} = 1$. Multiple independent Bernoulli trials with the same θ are called Bernoulli process. The related probability distribution is called Bernoulli distribution. Let X be a Bernoulli distributed random variable with parameter θ. For $\theta = 1$, the outcome of each trial in a Bernoulli process is always 0. Furthermore, it is always 1 for $\theta = 0$. Hence, the outcome of the system can be predicted perfectly in these two settings. The Shannon information of the system is

$$\mathrm{Inf}_{\mathrm{Sh}} = -(0 \cdot \log(0) + 1 \cdot \log(1)) = -(0 \cdot 1 + 1 \cdot 0) = 0. \tag{2.3}$$

This result matches the intuition that a system that is not at all surprising to the receiver does not produce any information. Speaking in terms of Shannon information, the system does not produce information. Using the sender and receiver illustration, we can also say that there is no information deficit created between the sender and the receiver.

On the other hand, for $\theta = 0.5$, we obtain

$$\mathrm{Inf}_{\mathrm{Sh}} = -(0.5 \cdot \log(0.5) + 0.5 \cdot \log(0.5)) = -(0.5 \cdot -1 + 0.5 \cdot -1) = 1. \tag{2.4}$$

Since the outcome of a Bernoulli trial is most unpredictable for $\theta = 0.5$, we obtain the highest possible Shannon information of 1 in this case. The information deficit between the sender and the receiver reaches its maximum. In this setup, the receiver is least likely to predict the output produced by the sender. Hence, the receiver is most surprised by the message of the sender.

2.2.2 Fisher Information

Another definition of information based on probability theory is the Fisher information (Fisher, 1925; Ly, Marsman, Verhagen, Grasman, & Wagenmakers, 2017). Let X be a random variable. Let $f(X,\theta)$ be a parametrized probability density function for X with parameter(s) θ. Fisher information measures the amount of information that the observable random variable X carries about the unknown parameter θ. The key idea is that it is easier to estimate the correct value of θ if X carries more information. Intuitively speaking, this means that a lot can be learned about θ by observing X if the Fisher information is high and only little can be learned about θ if the Fisher information is small.

Again, we can consider a Bernoulli experiment as an example. The Fisher information (Ly et al., 2017) of a Bernoulli distributed X parametrized by parameter θ is

$$\mathrm{Inf}_{\mathrm{Fi}}(\theta) = \frac{1}{\theta \cdot (1-\theta)}. \tag{2.5}$$

Hence, the Fisher information of X is small if θ is equal or close to 0.5 and large if θ is close to 0 or close to 1. Following the intuition from above, this means that we can quickly give a reasonable estimate about θ if θ is close to 0 or 1 and we have to observe more samples to have a reasonable estimate about θ if it is close to 0.5.

A normally distributed X can be considered as a second example. Let X be normally distributed with known σ^2 and unknown μ. The Fisher information of X is in this case

$$\mathrm{Inf}_{\mathrm{Fi}}(\theta) = \frac{1}{\sigma^2}. \tag{2.6}$$

Similarly to the Bernoulli example, a sharp density function (small variance σ^2) leads to high Fisher information, which means that we can learn quickly about θ if the variance is small. On the other hand, X carries only little Fisher information if the variance is high. Consequently, it takes more time to obtain a reasonable estimate of θ in this case.

A sample-based version of the Fisher information is called *observed information*. The Fisher information is simply defined as the expected value of the observed information. $\mathrm{Inf}_{\mathrm{Fi}}(\theta) = \mathbb{E}(\mathrm{Inf}_{\mathrm{ObsFi}}(\theta))$.

2.2.3 Kolmogorov Information

Both previously discussed measures of information are based on probability theory. According to Shannon, rare events carry more information, and according to Fisher, random variables from distributions with high kurtosis carry more information. Using probability theory as the basis for measuring information is, however, not the only possible way to measure the amount of information in a message.

The Kolmogorov information, also known as Kolmogorov complexity (M. Li & Vitányi, 2008), measures the amount of information of objects such as sequences of numbers based on the length of the shortest computer program which outputs the object when the program is executed. The intuition is that more complex objects require more complex descriptions (i.e., more complex computer programs) and that more complex objects contain more information. The concrete computer programming language is not crucially important as long as the same language is used to compare the information content of different sequences.

As an example, we can consider two sequences of numbers, both of which may contain digits from 1 to 3. Let sequence

$A = 313221131132313323112221131131$

and let sequence

$B = 123123123123123123123123123123.$

Both sequences have a length of $n = 30$ characters. Sequence A has been produced with a random number generator. Hence, we presume that the sequence does not follow a particular pattern. The shortest program (assuming a previously defined programming language) which can produce this random sequence is given in Algorithm 1.

Algorithm 1 Shortest program to generate sequence A

```
  print "313221131132313323112221131131"
```

Since the sequence is random, the only option is to store the complete sequence in the algorithm. Algorithm 1 consists of five characters for the print command, two characters for quotation marks, one white space character, and 30 characters for sequence A. With a total of 38 characters, it is rather long. Hence, the sequence has a high Kolmogorov complexity and therefore contains much Kolmogorov information.

Sequence B, on the other hand, follows a pattern. It can be constructed by repeating the sequence "123" ten times. Algorithm 2 contains a program which exploits this pattern to output sequence B.

Algorithm 2 Shortest program to generate sequence B

```
  print "123" * 10
```

The length of Algorithm 2 which produces sequence B is 16 characters long (five characters for the print command, two characters for quotation marks, three white space characters, one multiplication character, two characters to encode the number of repetitions, and three characters for the sequence 123). Since both algorithms use the same programming language, we can compare the Kolmogorov information of both sequences. With a length of its shortest producing algorithm of 16 characters, sequence B is less complex than sequence A since its shortest producing algorithm of 38. Hence, sequence A is more complex and therefore contains more Kolmogorov information than sequence B.

In general, it is difficult to find the shortest program which produces a given sequence (Bloem, Mota, de Rooij, Antunes, & Adriaans, 2014). Consider, for example, sequence

$C = 257829627945307727248226067259$

which looks similarly unstructured as sequence A. It can, however, be computed using the same programming language as used as in Algorithm 1 and Algorithm 2 with a program of length 13 and contains therefore even less Kolmogorov information than sequence B.[1]

Note that in Kolmogorov complexity, the probability that a message is observed is independent of its information content, which is not the case in Shannon information. We can, for example, use a Bernoulli distributed random variable X and can imagine a random process which produces the two discussed sequences A and B as output. Hence,

$\Omega = \{313221131132313323112221131131, 123123123123123123123123123123\}.$

We furthermore set

$\Pr(313221131132313323112221131131) = 0.9$ and

$\Pr(123123123123123123123123123123) = 0.1.$

[1] The producing algorithm is left as a riddle for the reader.

In this setting, it is much more likely to observe sequence A than sequence B.[2] Hence, the sequence A contains less Shannon information (e.i., it is more likely) but more Kolmogorov information (i.e., it is more complex) than sequence B.

2.2.4 Analysis

So far, we reviewed three definitions of information content. The Shannon and Fisher information functions are non-negative and additive which means that the sum of two mutually independent information pieces a and b equals to the information contained in the joint information piece $c = a \cup b$: $\text{Inf}(c) = \text{Inf}(a) + \text{Inf}(b)$. They furthermore both follow the idea that more information equals to more reduction of uncertainty.

All three definitions have in common that they are quantitative definitions. Shannon information specifies *how much* information is produced by a system. Fisher information specifies *how much* information is carried by a random variable. Kolmogorov information specifies *how much* information is contained in a sequence. They do, however, not discuss what properties information has to have. Discussing properties of information also does not make much sense in the case of Shannon, Fisher, and Kolmogorov since they use a purely technical meaning of the term 'information'. Information used in the context of these definitions does not have any *meaning*. Weaver (1949) writes with respect to the work presented by Shannon (1948): „First off, we have to be clear about the rather strange way in which, in this theory, the word 'information' is used; for it has a special sense which, among other things, must not be confused at all with meaning. It is surprising but true that, from the present viewpoint, two messages, one heavily loaded with meaning and the other pure nonsense, can be equivalent as regards information“.

As an example, we can consider the information contained in a well-written informative text (an encyclopedic entry, for example) and a document of the same length with a randomly generated sequence of letters. As discussed previously, a random sequence of equiprobable letters has the highest possible entropy. Hence, the latter text contains the highest possible Shannon information. Furthermore, it also contains the highest possible Kolmogorov information since the text does not follow any pattern. The latter text does, however, not contain any meaningful information. The amount of semantic information contained in a random sequence of letters is (most likely) 0 since the data cannot be interpreted and used in a meaningful way by a human.

The encyclopedic entry, on the other hand, has a lower amount of Shannon and Kolmogorov information since some letters are more likely to appear in a text in general. In English, for example, the letters 'e', 't', or 'a' are much more frequent than the letters 'x', 'q', or 'z'.[3] Furthermore, language contains structure and follows patterns. After a certain sequence of letters, it is almost clear which letter will appear next. Sometimes, even complete words can be predicted with a high probability. Following the example from the introduction, it is highly likely that the sentence „Donald Trump is the president of the

[2] Note that Ω contains only two possible outcomes, which are the two long sequences. The situation is different if we consider every character as one possible output.

[3] See http://pi.math.cornell.edu/~mec/2003-2004/cryptography/subs/frequencies.html, for example.

United States of“, is followed by the word „America“ assuming that the text is well-written as described before. Even though the text has a lower amount of Shannon and Kolmogorov information, it contains (more) semantic information.

This focus on the technical meaning and the limitation of the presented works has been reported in the literature many times (Floridi, 2010; Zhong, 2017). Weaver (1949) already noted that in any communication problems might occur at three different levels: *technical*, *semantic*, and *influential*, and that Shannon’s entropy is only concerned with the first technical level of information communication. More clearly, Weinberger (2002) writes: „’Standard’ information theory says nothing about the semantic content of information.“. Bar-Hillel (1969) points out that Shannon’s initial idea ’theory of information transmission’ was abridged to ’theory of information’ and later was converted into ’information theory’ (Gernert, 2006). Floridi (2011) writes: „After Shannon, MTC became known as information theory. [...] The term ’information theory’ is an appealing but unfortunate label, which continues to cause endless misunderstandings. Shannon came to regret its widespread popularity.“

Following von Weizsäcker (1985), who stated that „information is only that which is understood“, many researchers argue that what is estimated by Shannon information, Fisher information, or Kolmogorov information is actually not *information*, but merely *data*. Floridi states, for example, that information = (well-formed) data + meaning (Floridi, 2010). Hence, meaningless information is merely data and does not qualify as information. Gernert (2006) further writes: „Information exists as soon as the structure or the behavior of a recipient has been altered“.

2.3 Semantic Information

The observation that previous information definitions do not cover all relevant levels of communication but focus on the technical level led to the specification of *semantic information* (Bar-Hillel & Carnap, 1953; Floridi, 2010). Floridi (2010) defines information as follows:

Definition 3 (General Definition of Information). The element $\imath$ is an instance of information, understood as semantic concept, if and only if

1. $\imath$ consists of n data with $n \geq 1$,
2. the data are *well-formed*, and
3. the well-formed data are *meaningful*.

Definition 3 provides a qualitative definition of information in comparison to the three previously discussed definitions. It states which properties information has and not how much information is contained in the data at hand.

Note that Definition 3 defines the term *information* and does not need the adjective *semantic* because it distinguishes between data and information. All information with respect to this definition is semantic information, and ’information’ that is not meaningful (such as random sequences of numbers of randomly generated sequences of letters) is merely data. Simply speaking, the definition of information provided by Floridi (2010) can be summarized as *information = (well-formed) data + meaning*.

Furthermore, Definition 3 also states that the data has to be well-formed. Hence, it is requested that the data are put together according to the rules which are called *syntax*. Data that cannot be understood by a particular person because it does not follow a syntax which is known by this person is not meaningful for this person. Note, however, that another person might know the syntax which can be used to understand the information. Hence, whether or not data qualifies as information also depends on the knowledge of the receiver.

A simple example is a text written in a particular language. The text may contain meaningful information for a person who is able to understand the language and does not contain meaningful information for people who are not able to understand the language. We assume in the remainder of this thesis that the receiver is able to understand the information contained in the data at hand which means that we are only concerned with information in the sense of Definition 3.

2.3.1 Data - Carriers of Information

According to Definition 3, information is always made of data, which means that there can be no information without data. Therefore, we briefly discuss what ’data’ means in this context. Floridi argues that data are not a property of one single element in a system, but always the difference between two elements in the system. Consequently, a datum is ultimately reducible to a *lack of uniformity* (Floridi, 2010).

Definition 4 (General Definition of Datum). A datum = x being distinct from y, where x and y are two uninterpreted variables and the relation of ’being distinct’, as well as the domain, are left open to further interpretation.

Concrete instantiations for data are a black dot on a white surface, a higher and a lower charge in a battery, or two distinct symbols, A and B, in an alphabet. An example of a system that is not able to contain or store data is a system that is only able to take on one state. Let this state be named ’0’. The system does not contain data since the system is always in state ’0’. It is not possible to store any data or information (for example whether or not Donald Trump is the president of the U.S., or whether or not there have been Russian interferences). Similarly, a surface that contains only one pixel which can only take on one single state (for example ’white’) cannot be used to store information since its state cannot change. Which state the system or the pixel can take on is arbitrary. We can also call the state of the system ’1’ and the state of the pixel ’black’. No data can be stored as long as there is only one possible state.

This argumentation can be extended to systems or surfaces with multiple 'slots' for data. A system can have, for example, ten slots that can be filled and a surface can have ten pixels. If the system is only able to fill all slots with '0' (leading to the state '0000000000') and the surface can only fill all pixels with 'white', no data can be stored. In fact, it does not matter how many slots or pixels are available if all the slots have to be filled with the same symbol or color. The number of possible states the system and the surface can take on is still 1.

A system can only contain or store data if there is more than one possible state the system can take on. A classical computer, for example, can take on two possible states '0' and '1'. If the computer has only one slot available to store data, it takes on either state '0' or state '1'. The system is able to store data since it is possible to distinguish between two different states. Similarly, the surface contains data if pixels can take on different states. States can be, for example, 'white' and 'black' or a wide range of different colors.

Furthermore, all lack of uniformity can be considered to be data no matter whether the difference is intentionally or not. A random initialization of a computer hard drive with '0's and '1's is data in the sense of Definition 4. It is, however, no information in the sense of Definition 3.

Definition 3 and Definition 4 define what qualifies as information and data, respectively. Both definitions do, however, not specify how much information or how much data are contained in a message or a system. Contrary to the first three information definitions, Definition 3 and Definition 4 are qualitative and not quantitative. In the following, we provide quantitative definitions for the last two definitions as well.

2.3.2 Measuring the Amount of Data

Measuring the amount of data in a system can rather easily be performed based on the previously discussed prior work. As discussed above, a system has to have at least two different states (per slot) to store information. Therefore, it is the smallest possible system that can store information. Hence, it makes sense to use a unit which calculates how many binary systems (each of which has two states) are required to store an equivalent amount of data compared to the amount of data a system with an arbitrary number of possible states can store.[4]

The number of binary systems (or the number of binary slots in a single system) required to store a particular amount of data can be computed with the binary logarithm $\log_2(n)$ where n is the number of states the system at hand can take on. The unit of the respective measure is shannon[5] (Sh for short), or more commonly known as bit. We argued above that a system that can only take on one state cannot store data. Consequently, we would say that 0 binary systems are required to store the corresponding amount of data. This is perfectly reflected by the binary logarithm[6], since $\log_2(1) = 0$. The number of binary systems used to store an equivalent amount of data of a system with two states is trivially 1. Again,

[4] There are also other, less common units. The number of systems with 10 possible states required to store an equivalent amount of data, for example, is measured in 'hartley', which is also called 'ban' (Hartley, 1928).

[5] Similarly to other units relating information technology such as bit, erlang, and hartley, 'shannon' is also not capitalized.

[6] In fact, this is true for any logarithm.

number of states	amount of data in bits
1	0.000
2	1.000
3	1.585
4	2.000
5	2.322
6	2.585
7	2.807
8	3.000
9	3.170
10	3.322

Table 2.1.: List of the amount of data stored in a system depending on the number of possible states it can take on. The number of bits equals to the number of binary systems required to store the same amount of data.

the binary logarithm matches the intuition since $\log_2(2) = 1$. Similarly, trivial is to calculate the number of binary systems required to store four different states. Since we can express one system with four different states as two different systems, each of which only takes on two different states, the number of binary systems required to store the equivalent of this amount of data should be 2. The binary logarithm returns exactly this expected result ($\log_2(4) = 2$). The computation becomes more difficult and less intuitive if it is not possible to partition a system with many states into an equivalent integer number of binary systems. Storing the amount of data of a system with three different states, for example, requires $\log_2(3) \approx 1.585$ different binary systems. The math, however, still works in these cases. A system with three possible states and a system with two possible states can store $\log_2(3) \approx 1.585$ and $\log_2(2) = 1$, respectively. Together, they can take on six different states, which results in $\log_2(6) \approx 2.585$ binary systems. This is exactly the sum of binary systems required to represent the individual systems $\log_2(3) + \log_2(2) \approx 1.585 + 1 = 2.585 = \log_2(6)$. Table 2.1 lists the amount of data that can be stored by a system given the number of possible states of the system. Since computers are composed of elements, each of which can take on two different possible states, the amount of data a computer can store equals to the number of elements it is composed of.

As discussed above, any lack of uniformity can be considered to be data, no matter whether the difference is intentional or not. Hence, a system with a constant number of possible states contains always the same amount of data. Hence, the amount of data stored in a computer, for example, does not change since all bits are either in state ’0’ or state ’1’.[7] This observation confirms the need for a distinction between data and information since it is rather unintuitive that a randomly initialized system such as a computer contains information.

[7] We assume there that there are no additional states, which indicate ’invalid’ states, for example. In the case of a third state which indicates an invalid state, the system is not binary anymore but tertiary.

2.3.3 Measuring the Amount of Semantic Information

Previously, we discussed how the amount of data contained in a system can be measured. Similarly, we can compute the amount of data in a message. Let m be a message composed of characters from an alphabet of size 2. Let furthermore message m contain n symbols. According to the previous discussion, the message contains $\log_2(n^2) = n$ bits. The number of bits in a message does, however, not tell us anything about the amount of information contained in a message. As discussed above, a message with length m can contain a meaningless random sequence of characters. The message does not contain information in this case. On the other hand, it can also contain a sequence of characters that can be interpreted by a receiver and is therefore meaningful to the receiver. The amount of data contained in a message does therefore not provide an insight into the amount of information contained in the message. We discuss in the following a definition of semantic information proposed by Floridi (2010).

The key idea of semantic information, in general, is that it has some *meaning* which informally means that a receiver can interpret it. In other words, it means that knowledge *about something* can be extracted. We model in the following a system $\mathscr{S}$ about a receiver learns something. A system has a true state. In the following, we denote the true state of system $\mathscr{S}$ by s. The receiver does not know the true state. Hence, he or she has an information deficit. We model all the states which are potentially true from the receiver's perspective as set S. If the system $\mathscr{S}$ is a light bulb, for example, possible states might be 'on' and 'off'. Set S contains in this example both states 'on' and 'off' if the receiver does not know whether the system is in state 'on' or in state 'off'.

Definition 5 (Amount of Semantic Information). The amount of semantic information contained in $\mathfrak{i}$ about system $\mathscr{S}$ is defined as

$$\mathrm{Inf}_{Sem}(\mathfrak{i}) = |S| - |S_{\mathfrak{i}}| \tag{2.7}$$

where S models an information receiver's prior knowledge about all potentially true states of system $\mathscr{S}$ and $S_{\mathfrak{i}}$ models the information receiver's knowledge about the potentially correct states of $\mathscr{S}$ after receiving information $\mathfrak{i}$.

The key idea of the definition is to equate the semantic information of a message m and the amount of information a receiver of the message gains about the actual state of system $\mathscr{S}$. In other words, the amount of information in a message equals the amount of reduced information deficit about the true state of $\mathscr{S}$.

Let s be the state of the system $\mathscr{S}$ at hand. The most simple system is a binary system which can only take on two states as discussed above. The system can, however, also be more complex. In the following, we use a system with one integer-valued state variable for illustration ($s \in \mathbb{N}$). Let the true state, which is yet unknown to person p, be $s = 30$. Let furthermore s be restricted by a minimum value of 0 and a maximum value of 99. The boundaries of s are known by person p.

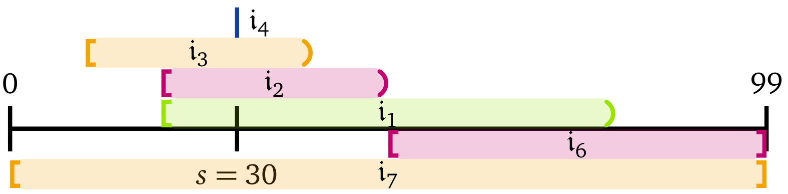


Figure 2.2.: Illustration of p's knowledge about the true state of system $\mathscr{S}$ which is $s = 30$. The intervals for $\mathfrak{i}_1$, $\mathfrak{i}_2$, $\mathfrak{i}_3$, $\mathfrak{i}_6$, and $\mathfrak{i}_7$ and the point for $\mathfrak{i}_4$ are displayed. $\mathfrak{i}_5$ is empty and therefore not displayed.

So far, person p knows that the true state of the system is $s \in [0, 99]$. The system can have 100 different states from person's p point of view, which represents in this example the amount of knowledge of person p about the system. Person p might receive one of the following messages:

1. $\mathfrak{i}_1 = s$ is larger than or equal to 20 and smaller than 80 ($20 \leq s < 80$); or
2. $\mathfrak{i}_2 = s$ is larger than or equal to 20 and smaller than 50 ($20 \leq s < 50$); or
3. $\mathfrak{i}_3 = s$ is larger than or equal to 10 and smaller than 40 ($10 \leq s < 40$); or
4. $\mathfrak{i}_4 = s$ equals to 30 ($s = 30$); or
5. $\mathfrak{i}_5 = s$ equals to 30 and equals not to 30 ($s = \emptyset$); or
6. $\mathfrak{i}_6 = s$ is larger than or equal to 50 ($50 \leq s \leq 99$); or
7. $\mathfrak{i}_7 = s$ is larger than or smaller than or equal to 50 ($0 \leq s \leq 99$).

The corresponding possible states $S_{\mathfrak{i}_i}$ from person p's point of view are illustrated in Figure 2.2.

Information $\mathfrak{i}_1$ reduces the uncertainty about the state of the system of person p by 40 since 40 states can be excluded by p given information $\mathfrak{i}_1$. According to Definition 5, the amount of semantic information contained in $\mathfrak{i}_1$ is

$$\mathrm{Inf}_{\mathrm{Sem}}(\mathfrak{i}_1) = |S| - |S_{\mathfrak{i}_1}| = 100 - 60 = 40. \tag{2.8}$$

Information $\mathfrak{i}_2$ restricts the set of remaining possible states more than inf_1 since all states in $S_{\mathfrak{i}_2}$ are also contained in $S_{\mathfrak{i}_1}$, but not vice versa which means that $S_{\mathfrak{i}_2} \subset S_{\mathfrak{i}_1}$. Hence, inf_2 reduces the uncertainty of p about the system more than inf_1. This is also reflected by Definition 5, since $\mathrm{Inf}_{Sem}(\mathfrak{i}_2) = 100 - 30 = 70$. Information $S_{\mathfrak{i}_3}$ is not a subset of $S_{\mathfrak{i}_1}$ but has the same size $S_{\mathfrak{i}_2}$. Hence, $\mathfrak{i}_3$ has the same amount of information according to Definition 5 as $\mathfrak{i}_2$. Even though inf_3 is not implied by inf_1, we can say that it contains more information than inf_1 since the number of remaining possible states is smaller.

Information $\mathfrak{i}_4$ restricts the space of possible states to only one remaining possible state $S_{\mathfrak{i}_4} = \{30\}$. Consequently, the amount of information carried by $\mathfrak{i}_4$ is: $\mathrm{Inf}_{Sem}(\mathfrak{i}_4) = 100 - 99 = 1$. Since it describes the state perfectly, nothing more can be learned about s. Or in other words, the information deficit has been reduced to 0 by $\mathfrak{i}_4$. However, $\mathfrak{i}_5$ seems to contain even more information about state s than $\mathfrak{i}_4$ since $\mathrm{Inf}_{Sem}(\mathfrak{i}_5) = 100 - 0 = 100$ even though $\mathfrak{i}_5$ is always wrong (it is a contradiction) no matter what the actual state s is. This problem is called Bar-Hillel-Carnap paradox (Floridi, 2010).

Definition 5 implements the idea that a piece of information is informative if it reduces the space of remaining states. The fewer states are left, the more informative an information is. This principle is also known as inverse relationship principle (IRP) (Floridi, 2010). A contradiction, however, has 0 remaining states as a consequence. Hence, a contradiction would be most informative. A simple solution to the paradox is to not consider incorrect information as semantic information. This leads to the theory of *strong* semantic information (Floridi, 2010). The theory of strong semantic information adds an additional requirement to Definition 3 where we defined what qualifies as semantic information. In addition to the requirement that data has to be meaningful, it also requests that a meaningful datum is correct in the sense of 'does not contradict with the true state of the system at hand'. Incorrect information is also called misinformation or disinformation, depending on whether the information is unintentionally incorrect (misinformation) or intentionally incorrect (disinformation). We have already encountered this further requirement in the out-of-scope section (see Section 1.4) in which we stated that truthfulness is out-of-scope of this thesis and that we assume that all information which we observe is true. By assuming that all observed information is true, we are able to avoid the pitfalls of the Bar-Hillel-Carnap paradox. We can include this restriction by changing Definition 5 to

Definition 6 (Amount of Strong Semantic Information). The amount of strong semantic information contained in $\mathfrak{i}$ is defined as

$$\mathrm{Inf}_{\mathrm{Sem}^*}(\mathfrak{i}) = |S| - \frac{|S_{\mathfrak{i}}|}{|S_{\mathfrak{i}} \cap s|} \tag{2.9}$$

where s is the true state of the system at hand and S and $S_{\mathfrak{i}}$ are defined as in Definition 5.

Contradictions lead in this equation to a division by 0 (since $|S_{\mathfrak{i}} \cap s| = 0$ if $s \notin S_{\mathfrak{i}}$) which can be interpreted as an invalid use of the function.

Information $\mathfrak{i}_6$ is not a contradiction by itself since it is true for other states in which the system can be in. However, it contradicts with the true state s of the system is this example and does, similarly to $\mathfrak{i}_5$ not qualify as information. Last but not least, information $\mathfrak{i}_7$ is a tautology. Hence, it is always true no matter in which state the system actually is. Since we do not learn anything about the system state, we would intuitively say that $\mathfrak{i}_7$ does not contain any information. Definition 5 works in this setting as well since $\mathrm{Inf}_{Sem}(\mathfrak{i}_7) = 100 - 100 = 0$.

As a more concrete real-world example, we can consider the U.S. presidential election from the introduction. Let the set of possible outcomes of the election be $S = \{\mathrm{Clinton}, \mathrm{Trump}, \mathrm{Johnson}, \mathrm{Stein}, \mathrm{McMullin}\}$.[8] Similar to the computation above, we can, for example, compute the amount of information in the mes-

[8] For simplicity, not all candidates are contained in the set.

sages that either Clinton or Trump won. The message that Johnson won contradicts with the true state of the system and is therefore not considered to be information. Similarly, the information that Clinton *and* Trump won is a contradiction since it is always incorrect, no matter what the actual result of the election was.

The example can also be used to illustrate that prior beliefs about the system are not considered by Definition 5. If a person receives the information that either Trump or Johnson won, the person could conclude with a high certainty that Trump won. This estimation is only possible if the person has the prior knowledge that it is much more likely that Trump wins instead of Johnson.

Similar to Shannon and Fisher information, the quantitative definition of semantic information is also non-negative, additive, and follows the idea that information reduces uncertainty. More information reduces the uncertainty more than less information. The inverse relationship property also states that the amount of semantic information is 0 for information which does not exclude any possible states. Hence, it is exactly the opposite behavior of a measure in measure theory, which is 0 only for empty sets.

2.3.4 Analysis

After discussing three quantitative information definitions which are concerned with the first technical level of communication, we have discussed now a qualitative and two quantitative definitions of information which are concerned with the second, semantic level of communication. The crucial difference between the technical definitions and the semantic definitions is the fact that the data has to be meaningful to be considered valuable by the semantic information definitions. In the natural number example, the data was meaningful because the receiver of the data was able to update its knowledge about a given system. In other words, the information in the data was linked to something in the world at hand. In the second example of the US presidential election, the grounding is more obvious since the elements in the set are representative of people in the real world. However, the example works as well if we replace the meaningful names in the set with arbitrary symbols.

None of the definitions is, however, concerned with the third level of information which is concerned with the impact information has to a receiver. Similarly to the fact that syntactic information (data) does not have to contain any semantic information, semantic information does not have to contain *important* information necessarily. We illustrate this gap in the next section with a lottery game.

2.4 Information Importance

So far, we discussed measures that estimate information on the technical and the semantic level. Semantic information theory discusses what properties information has, and furthermore provides ideas on how to measure the amount of semantic information contained in data. Works on semantic information are, however, not concerned with the question of how important semantic information is. The key idea of *pragmatic information* is to measure the information by its impact on a receiver. Hence, pragmatic in-

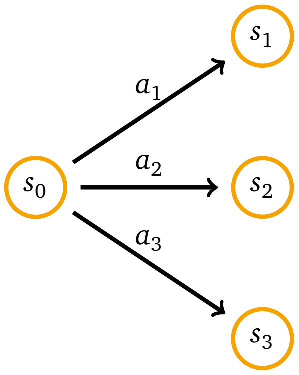


Figure 2.3.: Illustration of a person's available actions in state s_0 and the resulting states s_1, s_2, s_3.

formation is concerned with the third level of communication as defined by Weaver (1949). We illustrate this in the following with a lottery game.

Playing the lottery

Let p be a person who may play a lottery game. If person p decides to play the lottery, he or she has to choose a sequence of numbers. The player wins the main prize if the lottery game produces the same sequence of numbers.

The behavior of p is modeled by a *policy* π_p. A policy determines which action a person performs given a specific state of the world the person lives in. For simplicity, we omit to indicate to which person a policy belongs and write π in the following. The policy specifies, for example, whether or not the person plays in the lottery and which numbers the person chooses if he or she plays in the lottery. Let S be the set of states in which person p's world can be in. We denote individual states by $s_i \in S$. The actions a person is able to perform depend on the current state. Some actions can only be performed in specific states. Let A_{s_i} be the actions which can be perform by person p in state s_i. We omit for simplicity this fact and denote the available actions by set A and implicitly keep in mind that set A depends on the current state. The policy π according to which person p acts can be described as a function which maps from states to actions. Hence, we can write π as function $\pi : S \rightarrow A$ which maps from every possible state to an action. π can also be formulated probabilistically as $\pi(a|s_i), a \in A, s_i \in S$ which models the probability that person p performs action a given that the person is in state s_i.

Figure 2.3 illustrates a situation in which person p is currently in state s_0. Person p can perform in this state three different actions a_1, a_2, and a_3 which lead to the states s_1, s_2, and s_3, respectively. Important to note is that person p does not necessarily know how the resulting states look like. In the lottery example, the different actions may refer to different numbers person p can choose from. He or she does, however, usually not know which action (i.e., number selection) leads to winning the main prize.

Since we want to investigate the impact of information on the behavior of a receiver, the last missing piece is a piece of information. Let $\mathfrak{i}_i$ be information pieces whose importance for person p has to be estimated. The policy π of a person is not constant but changes over time. In particular, the behavior of a person changes if the person consumes new information (such as information piece $\mathfrak{i}_i$). In driving lessons, for example, a student driver learns to brake if an obstacle occurs on the street. The behavior of the learner (his/her policy) π is modified in the training such that the learner does not crash into obstacles. The information provided by the driving instructor changes the behavior of the student driver (given that the information is correctly perceived and understood, which we assume to be always the case in this thesis). Generally speaking, learning means nothing else but a change of behavior (Bergius, 1971; Hilgard, 1948).

Hence, we define that $\mathfrak{i}_i$ is an information which leads to person p performing action a_i. Let us furthermore define that person p performs action a_1 if he or she does not receive any new information.

2.4.1 Behavior-focused Importance of Information

We formulate the first information importance definition in terms of the amount of behavior change a piece of information entails.

Definition 7 (Behavior-focused Importance of Information). Formally, for a person p with policy π and prior knowledge $\mathfrak{I}$, we define the importance of $\mathfrak{i}$ as

$$\mathrm{Imp}_p(\mathfrak{i}) = d(\pi_{\mathfrak{I}\cup\mathfrak{i}}(a|s), \pi_{\mathfrak{I}\setminus\mathfrak{i}}(a|s)), \tag{2.10}$$

where d is a distance between two probability distributions, $\pi_{\mathfrak{I}\cup\mathfrak{i}}$ is the policy of person p *with* knowledge of information $\mathfrak{i}$, and $\pi_{\mathfrak{I}\setminus\mathfrak{i}}$ is the policy of person p *without* knowledge of information $\mathfrak{i}$.

A concrete instantiation of distance d can, for example, be the Kullback-Leibler distance (Endres & Schindelin, 2003; Kullback & Leibler, 1951). The Kullback-Leibler distance is based on the Kullback-Leibler divergence which is, however, not a proper metric (Endres & Schindelin, 2003).

Informally, this definition states that more important information changes the behavior of a person more substantially, or in more states, or both. Minor important information does not have such a substantial impact on the conduct of the affected person. This definition is close to pragmatic information, which is concerned with the impact a piece of information has on a receiver.

Unfortunately, Definition 7 does not always conform with an intuitive notion of information importance. Consider, for example, that the information $\mathfrak{i}_i$ in the lottery example contains information about the correct lottery numbers of next week's lottery game.

A person who plays the lottery game every week by picking ten arbitrary numbers usually looses since it is very unlikely to pick the correct numbers. If the person receives the correct lottery numbers of next week's lottery game (provided by an oracle), he or she will change the behavior and will choose the ten provided numbers instead of 10 arbitrary numbers to win the lottery. The behavioral change induced by the provided information in the person p's policy π is rather small since it affects only one state s_i. In the second-next week's lottery game, the information is useless. Furthermore, the person still plays

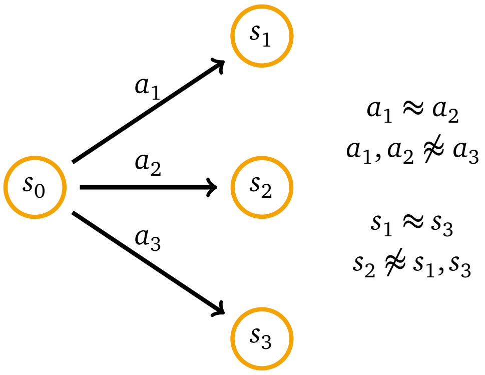


Figure 2.4.: Illustration of a person's available actions in state s_0 and the resulting states s_1, s_2, s_3. In this illustration, actions a_1 and a_2 are similar to each other but dissimilar to action a_3. States s_1 and s_3 are similar to each other but dissimilar to state s_2.

the lottery and just selects different numbers. The performed action with the knowledge of the correct numbers $\pi_{\mathfrak{I}\cup \mathrm{i}}$ is therefore not very different from the usually performed action without knowing the correct numbers $\pi_{\mathfrak{I}\setminus \mathrm{i}}$. However, winning the lottery can have a huge impact on the life of the person due to the received money. Therefore, the importance of this particular information should be considered to be rather high than low. Hence, the information importance definition provided in Definition 7 does not match the intuition in this example.

The example is illustrated in Figure 2.4. The person's current state is s_0. The person's default action is modeled by a_1, which represents the action of picking some arbitrary numbers without the knowledge of the correct lottery numbers. In this situation, person p will go (most likely) to state s_1 in which the person does not win the lottery. With the knowledge of the correct numbers, the person will take action a_2, which leads to winning the lottery. Both actions a_1 and a_2 are similar even though they result to very different states s_1 or s_2. A more dissimilar action a_3 (e.g., not playing the lottery at all) leads to state s_3 which is very similar to state s_1 (in both states the person does not win the main prize). Hence, similar actions can lead to dissimilar states, and dissimilar actions can lead to similar states.

Estimating information importance based on the similarity of actions appears not to match the natural intuition of information importance, which leads to the following definition of importance which focuses on states instead of actions.

2.4.2 State-focused Importance of Information

Intuitively, we would like a definition of importance that does not measure the change of conduct of a person but the change in the course of life of the person. This is modeled by Definition 8.

Definition 8 (State-focused Importance of Information). The importance of information equals to the difference of the course of life caused by the behavioral change of the audience. Formally, for a person p with policy π and knowledge $\mathfrak{I}$, we define the importance of i as

$$\mathrm{Imp}_p(\mathrm{i}) = d(\Pr(s|\pi_{\mathfrak{I}\cup \mathrm{i}}), \Pr(s|\pi_{\mathfrak{I}\setminus \mathrm{i}})) \tag{2.11}$$

where d is a distance between two probability distributions, $\pi_{\mathfrak{I}\cup\mathfrak{i}}$ is the policy of person p *with* knowledge of information $\mathfrak{i}$, $\pi_{\mathfrak{I}\setminus\mathfrak{i}}$ is the policy of person p *without* knowledge of information $\mathfrak{i}$, and $\Pr(s|\pi)$ models the probability of being in state s given a policy π.

Definition 8 follows a statement made by Immanuel Kant already about 200 years ago: „[Die Wichtigkeit eines Erkenntnisses] beruht auf der Größe oder Vielheit der Folgen. Je mehr oder je größere Folgen ein Erkenntniß hat, je mehr Gebrauch sich von ihm machen läßt, desto wichtiger ist es" (Kant, 1800) which roughly translates to: „The importance of an insight/information is based on the amount or multiplicity of the consequences. The larger the number of consequences or the more substantial the consequences of an insight/information are, the more useful it is, the more important it is".

Definition 8 does not have the discussed issue of Definition 7 since it focuses on the states a person p will be in and not on p's policy. If we consider the lottery example again, this means that a piece of information is considered important if it leads to winning the main price even if the behavioral change of the person is small and it is not considered important if it does not lead to winning the main price even if the behavior change is large.

Related to the presented information importance definition is the *value of information* (Marshak, 1954), which is defined as the amount of money a rational decision-maker would maximally pay to obtain the information. Hence, it represents the value of the information with respect to the improvement of the decision maker's decision outcome. A requirement to apply this idea is that the decision-maker can compute the outcome (or an expected outcome) of a decision. If we consider the lottery numbers again, a rational decision-maker would maximally spend x amount of money, for example, to obtain the information which will lead to a win of x money. If the price for the information is higher, it does not make sense for the player to obtain the information since the return of investment is negative. Contrary to Definition 8, Marshak (1954) focuses on the improvement on an agent's decision making abilities. Definition 7 and Definition 8 estimate the importance of information in a broader sense since no concrete decision making situation is assumed. We follow the idea that learning always leads to a change of behavior (Bergius, 1971; Hilgard, 1948), and asses the impact of this behavioral change in two different ways. Hence, information is not something that is actively used by a person but can also influence its behavior in more subtle ways. Similarly, Weinberger (2002) define the value of information as its usefulness in making an informed decision. Wilkins, Lee, and Berry (2003) extend the idea of value of information to the *value of an alert*. The value of an alert takes the value of information into account but weighs it against the costs of interrupting a user who might do something more important currently. Interrupting a person while he or she is driving a car, for example, should usually be avoided since the cost of confusing the driver might be large. The driver should only be interrupted if the information is critical. We assume in this work that we do not interrupt a user from current activities and focus on estimating information importance. Hence, the value of an alert is out-of-scope of this thesis. However, if this work leads to applications which can, for example, notify end-users via push-messages on a mobile phone, estimating the costs of interrupting a user might be considered as well.

2.4.3 Properties and Implications

We now discuss some properties and implications of the importance definition provided in Definition 8.

Information Importance is Personalized

The first inherent property of information importance according to Definition 8 is that information importance can only be estimated in a meaningful way with respect to an audience because the importance of information depends on the receiver of the information. Personalization fits the intuition that some information might be more or less important to different people. The results of local elections, for example, might not be very important for people living far away from the voting area but can be very important to directly affected people. It has also been reported, for example by P. V. S. and Meyer (2017), that users prefer to read information according to their personal needs. Furthermore, the resulting texts differ if humans are asked to write a summary of a text which contains the most important information contained in the text. This fact is another indication than information importance should indeed be estimated with the reader in mind.

Information Importance is Invariant of Redundancy

Definition 8 compares the states reached by a person with and without a particular information piece. More formally, we proposed to compare $\Pr(s|\pi_{\mathfrak{I}\cup\imath})$ and $\Pr(s|\pi_{\mathfrak{I}\setminus\imath})$. Since we use $\mathfrak{I}\setminus\imath$, it is not relevant for the importance of an information piece whether or not it is already known by person p. This property fits the intuition that we, for example, say that the information that Donald Trump won the U.S. Presidential Election is important information even though many people already know this information.

Information Importance is Time-dependent

Due to its definition based on the states reached by a person, information importance also depends on the time when a piece of information is consumed. The very same information piece $\imath$ can be important for person p at a point in time t_1, which corresponds to a state s_i and unimportant at time t_2, which corresponds to another state $s_i \neq s_j$. Let, for example, $\imath$ the lottery numbers of a particular lottery game which happens at time t. If person p receives the information before the lottery starts, it is important because the information changes the course of life of p substantially. If the information is received after the lottery happened, the course of life of p may not differ a lot since the information only impacts states which cannot be reached anymore by p. If s_i corresponds to the state in which person p wins the main prize, and the lottery is already finished such that p cannot win the price anymore, $\Pr(s)$ will be 0 (or almost 0) no matter if p has the information $\imath$ about the correct lottery numbers or not.

Information Importance is Morally Neutral

According to Definition 8, the importance of information is defined as the distance between two probability distributions over states reached by the information receiver. The definition does, however, not measure whether the impact on the receiver's life is positive or negative. It is not assessed if person p reaches states, which makes him or her happier or if an information piece leads to a fulfilling life. Hence, it is independent of any moral assessment. Information is defined as important even if it negatively impacts the life of a person.

The definition might be modified to model if information is *desirable* such that receiving more desirable information is valued. Even though this line of thought is interesting, we consider it as out-of-scope of this thesis since it is even more challenging to assess whether or not an impact on the curse of life of a person is desirable. Furthermore, it is perhaps even more challenging to find or generate data for this definition of information which we will need in Part II of this thesis where we discuss how machine learning can be used to estimate information importance.

Information Importance is Invariant of Receiver's Desires

Similar to the previous point, information importance is also invariant of the receiver's desires. Information can be unimportant even if a reader likes a lot or disagrees with the information he or she reads. For example, a person might like to read the information that his or her favorite soccer team won a match. If this information does not change the course of life of the person a lot, it is considered to be unimportant according to Definition 8. Hence, information importance as defined above is also not subject to filter bubbles. Furthermore, information which is consumed by a receiver for entertainment purposes is also not considered important. For instance, consider the broad coverage of royal weddings in the television. Even though royal weddings do no longer influence the lives of many people, they are still followed by many people. Hence, royal weddings are unimportant according to Definition 8, but it might be interesting to many people. Hence, we distinguish between important and interesting information in this thesis.[9]

Information Importance is Non-additive

Definition 8 only provides a definition for an individual information piece $\mathfrak{i}$ given a set of already available information $\mathfrak{I}$. Hence, the importance of information can change if the set of already available information changes. Consider, for example, that we provide a person with correct lottery numbers of next week's lottery not in plain text but in a form which has been encrypted twice with two different encryption keys contained in information $\mathfrak{i}_1$ and $\mathfrak{i}_2$. Providing the person with only one of the keys does not change the person's behavior since the person is not able to decrypt the lottery numbers. The importance of each individual encryption key is therefore 0: $\text{Imp}(\mathfrak{i}_1) = \text{Imp}(\mathfrak{i}_2) = 0$. This result is intuitively correct since each information piece on its own does not have a value for the recipient. Only providing

[9] We discuss different purposes of reading in more detail in Section 3.1.1.

both keys jointly has an impact on the behavior. Consequently, the importance of both keys together should be considered to be larger than 0.

It is possible to extend Definition 8 to model not only the importance of individual information pieces but also to model the importance of sets of information pieces. Definition 8 can be changed from

$$\mathrm{Imp}_p(\mathfrak{i}) = d(\Pr(s|\pi_{\mathfrak{I} \cup \mathfrak{i}}), \Pr(s|\pi_{\mathfrak{I} \setminus \mathfrak{i}})) \tag{2.12}$$

to

$$\mathrm{Imp}_p(\mathfrak{I}_2) = d(\Pr(s|\pi_{\mathfrak{I}_1 \cup \mathfrak{I}_2}), \Pr(s|\pi_{\mathfrak{I}_1 \setminus \mathfrak{I}_2})). \tag{2.13}$$

where $\mathfrak{I}_2$ models a set of information that is received instead of an individual information piece.

For two encryption keys $\mathfrak{I}_2 = \{\mathfrak{i}_1, \mathfrak{i}_2\}$, we would like to obtain $\mathrm{Imp}(\mathfrak{i}_1 \cup \mathfrak{i}_2) > 0$ as discussed above. Therefore, we get $\mathrm{Imp}(\mathfrak{i}_1 \cup \mathfrak{i}_2) > \mathrm{Imp}(\mathfrak{i}_1) + \mathrm{Imp}(\mathfrak{i}_2)$ which means that Imp() is not additive. Information pieces can therefore show synergy effects which means that two information pieces can have a larger joint effect on a person than one of the two individual pieces. Similarly, two information pieces can have an eliminating effect. Providing a person with only one of the two information pieces from $\mathfrak{I}_2$ may lead to a change in the life of the person. Hence, it would be considered to be important. Providing a person with both information pieces may, however, lead to no changes. We get in this case $\mathrm{Imp}(\mathfrak{i}_1 \cup \mathfrak{i}_2) < \mathrm{Imp}(\mathfrak{i}_1) + \mathrm{Imp}(\mathfrak{i}_2)$.

Information Importance is Invariant of the Context it Appears in

Definition 8 states that the importance of information depends and only depends on its impact on a receiver. Hence, it is not relevant in which context a piece of information appears. The same information piece has the same importance in every context as long as the receiver (and its prior knowledge) does not change. It does, for example, no matter in which context the information that Donald Trump won the election appears in.[10] Let $\mathfrak{i}$ be the information of Donald Trumps winning the election and let d_1 and d_2 be two text documents in which the information appears in (informally: $\mathfrak{i} \in d_1$ and $\mathfrak{i} \in d_2$). The importance of the information $\mathfrak{i}$ is the same in both documents (contexts) since the definition of information importance does not make any use of d_1 or d_2.[11]

This observation is important for the next chapter, in which we discuss how summarization is related to information importance estimation and how they differ. In summarization, the terms *important* information and *main* or *major* points of a text document are used synonymously. We will differentiate

[10] Recall that we always assume that the receiver of the information understands and trusts the information and that all observed information is correct.

[11] We discuss in Section 4 in more detail what we exactly mean with the term 'context'

between both concepts, which is important to understand the focus of this thesis estimating information importance without being focused on what the major points of a document are. We discuss this in more detail in Chapter 3.

2.4.4 Examples

The provided importance definitions are rather abstract, and it is rather unlikely that they can be used directly to estimate the importance of information since it is not possible in practice to measure the distance between two life stories. The definitions are meant to be rather abstract ideas to enable humans to discuss more precisely about information importance similar to previous work discussed how information can be defined (Floridi, 2010). One example of a more precise definition is the discussion about the implications of the definition provided in Section 2.4.3. Such a discussion is not possible without a rather formal definition.

To mitigate the abstractiveness of the provided information importance definition, we provide in the following more concrete examples of how the definition can be interpreted to provide a better understanding.

U.S. Presidential Election

As a first example, we want to discuss the motivational example presented in Chapter 1. According to Chapter 1, the information that Donald Trump won the election is more important than the fact that the U.S. Congress certified the results of the election on January 6. This estimation is accurately explained by Definition 8 since the outcome of the election affects the curse of life of many people. People discuss the result and the potential impacts of the election much more frequently than the certification of the results. The behavior of many people in the United States is affected by the laws and executive orders put into place by the president. Furthermore, the behavior affects many people all over the world who have to prepare for the effects of trade wars, for example. The information about the certification of the results, on the other hand, does not affect the behavior of many people. It does not matter whether or not people know that the results have been certified on January 6.

Scientific Publications

The provided definition of importance can also be applied to scientific publications. Important publications are publications that have a big impact on many researchers. Hence, the importance of a scientific paper by the amount of change it generates in the scientific community and more broadly to society in general. Papers such as the 2018 ICML Test of Time Award winner „A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning“ (Collobert & Weston, 2008) are, according to Definition 8, important papers since they influenced the behavior of many researchers. Without the information included in this paper, many research would have never worked on end-to-end deep learning architecture. On the other hand, contributions that do not have a big impact on researchers or the society are considered to be rather unimportant to said researchers and members of this society.

However, Flexner (1939) already wrote about the usefulness of apparent 'useless' information. He writes that information (i.e., a scientific discovery) can be useless at the time of the discovery but very useful later. As an example, consider the experiments of Heinrich Hertz with electromagnetic waves (Faccio, Clerici, & Tambuchi, 2006). At the time of the discovery, the experiments were useless since the knowledge generated by Hertz did not have any application. Hence, it was unimportant to people living at this time. However, Marconi contributed to the development of wireless telegraphy based on Hertz's discoveries later, which turned out to be a revolutionary technology. The contributions by Hertz are therefore important to people who lived at this time and later. Hence, we also conclude that estimating the importance of scientific contributions is a hard problem and might only be possible with a higher accuracy later in time.

Machine Learning

In supervised machine learning (discussed in more detail in Part II of this thesis), machines train to perform well at a task by observing training examples. Training instances that do not change the behavior of the machine learning model are unimportant since they do not impact the behavior of the model. Hence, it is not important whether or not the learning algorithm observes unimportant training information. Training examples that change the behavior of the model substantially are, on the other hand, considered to be important. Zintgraf, Cohen, Adel, and Welling (2017), for example, analyze which areas of images are important for the prediction of machine learning models. They test how the prediction of a classifier changes if specific parts of an image are hidden. Areas are considered to be important if the classification of the machine learning model changes when the area is not visible to the model. In these examples Definition 7 is more appropriate since the models are rather static.

P ≠ NP

In computer science, it is widely assumed that $P \neq NP$ is correct (Cook, 2000), meaning that there are problems which are inherently more difficult to solve than to verify. The consequences of receiving the correct information that $P = NP$ is correct would be huge since difficult NP-complete problems are fundamental in many fields. An efficient solution for a problem such as 3-SAT, for example, would break many existing cryptographic systems. All systems relying on the assumption that integer factorization is a difficult problem would have to be replaced due to insecurity. Hence, the information that $P = NP$ is correct would perhaps foster much more research in areas that are concerned with alternative approaches for encryption. However, receiving the information that $P \neq NP$ would not have such a big impact due to the widespread belief that $P \neq NP$ is correct. According to Definition 8, the information would be rather unimportant since it does not have a big impact on the behavior of people.

2.5 Summary

In this chapter, we first reviewed works that are concerned with defining the term ’information’. Several researchers have introduced formal methods to measure the amount of information. Hence, ’information’ has been used with different meanings in different research areas and different periods.

We discussed three well-known quantitative information definitions: Shannon information (Shannon, 1948), Fisher information (Fisher, 1925), and Kolmogorov information (M. Li & Vitányi, 2008). All three definitions use a rather technical meaning of information. Floridi (2010) has later argued that data might be a better fitting term for these works since they are not concerned with the meaning of information. Random sequences of symbols, for example, have a high information value according to Shannon information and Kolmogorov complexity but do not contain any meaningful information. We continued to discuss a qualitative information definition presented by Floridi (2010) called semantic information. The key idea here is only to consider data as information if it is meaningful, which means that a receiver can interpret it. This work is crucially different from the three previously discussed information definitions since it focuses on the receiver of the information and its interpretation of the information. The basic idea is that the distance of a true state of a system and the description of the system contained in a proposition can be used to estimate the amount of semantic information. We furthermore discussed how the amount of data and semantic information can be measured. They analyzed that the amount of semantic information is, however, also not indicative of the importance of information.

Hence, we extend prior work by providing two definitions of information importance. The first definition states that the importance of information should match the impact on the behavior of the recipient. This definition, however, does not perfectly match an intuitive definition of information importance. To solve this problem, we proposed a second definition which states that the importance of information should match its impact on the course of life of the recipient.

We discussed the logical implications of the provided information importance definition. Additionally, we provided some examples which illustrate applications and analyzed implications of the rather theoretical definitions. We argued that the proposed definition matches the common intuition of information importance.

Definitions are neither right nor wrong but should better be considered to be useful or not useful. Similar to the various definitions of information and their different properties, we expect that further definitions of information importance exist. We hope that the provided formal definition and the discussion of the implications can help to refine the intuition of information importance in future work.

After having a formal definition of information importance, we are well equipped to discuss how information importance estimation capabilities can be assessed in the next chapter. We use the formal information importance definition proposed in this chapter to define how optimal summaries look like in the next chapter, which is the first application of the proposed definition.

3 Evaluating Information Importance Estimation Abilities

In this thesis, we aim at laying the foundations for context-free information importance estimators. In Chapter 2, we have already created a common ground for the meaning of information importance. Before we discuss in Chapter 4 more formally what exactly *context-free* information importance estimators are and why we need *context-free* information importance estimators, we first investigate how we can measure the quality of information importance estimators in general by answering research question two: **How can information importance estimation abilites be assessed?**

In Section 3.1, we discuss the most prominent scientific area in which information importance estimation abilities are required, namely *(automatic) summarization* (Mani, 2001; Nenkova & McKeown, 2011). The goal in automatic summarization is to condense a text document into a summary of said document. A good summary should contain the most important information which is contained in the source documents. Hence, summarization systems need good information importance estimation abilities. In Section 3.1.1, we discuss the two main purposes of reading briefly. In Section 3.1.2, we review the literature to identify important properties of good summaries. Most relevant to this work is the requirement for good summaries is to contain the most important information contained in a summarized source document. The fact that information importance is a central aspect of automatic summarization makes it surprising that there has been no definition of information importance proposed so far. In Section 3.1.3, we discuss a formal definition of summarization.

Based on the definition of information importance in Chapter 2 we are able to discuss optimality of summaries in Section 3.2. Concretely, we formally specify how the *optimal personal summary* for given text documents and a particular person p looks like in Section 3.2.1, and we generalize the personal definition to arbitrary audiences in Section 3.2.2. This clear definition of an optimal summary was not possible without a formal definition of information importance. We observe a close connection between the newly presented definition of optimal summaries and the well-known Pyramid evaluation method (Nenkova & Passonneau, 2004). In Section 3.3.1, we focus on a special kind of summarization called extractive summarization. The key idea of extractive summarization is to extract and concatenate text snippets from input documents instead of writing a new text from scratch to create summaries. This subfield of summarization is very appealing to study information importance estimation since other difficult problems such as generating fluent and grammatically correct sentences is not a concern. We also describe a simple approach for extractive summarization based on greedy algorithms.

In Section 3.3, we define ’extractive summarization’, which is an often used solution approach for automatic summarization. The key idea, which is formalized in Section 3.3.1, is to summarize documents by extracting sentences from source documents. The extracted sentences are then concatenated to generate a summary. In Section 3.3.2, we discuss a simple solution approach for extractive summarization that selects sentences greedily.

In Section 3.4, we discuss shortcomings of using automatic summarization if we are mainly concerned with information importance estimation. This leads to the conclusion that automatic summarization is not perfectly appropriate to evaluate information importance estimation abilities.

In Section 3.5, we propose more appropriate evaluation setups. Specifically, we propose to evaluate a system's information importance estimation abilities with rankings in Section 3.5.2, preference prediction in Section 3.5.3, and regression tasks in Section 3.5.1.

We summarize this chapter in Section 3.6.

3.1 Foundations of Automatic Summarization

We discuss in this section the task of (automatic) summarization as a means of evaluating the information importance estimation capabilities of machines (and humans). The goal in summarization is to reduce the size of a set of information nuggets such that only the most important information remains. Hence, it is necessary to estimate the importance of information to perform well in this task. Without good information importance estimation capabilities, it is not possible to distinguish between important and less important information. An example of summarization is the situation described in the introduction in which a journalist has to reduce a set of three information nuggets to a set of one information nugget. A general goal of this thesis is to foster the development of machines that are able to support humans in managing information overload. Automatic summarization is one way to approach this goal.

To become more precise what summarization is about, we first review the two purposes of reading in Section 3.1.1 and review the literature to extract and condense several definitions of summarization in Section 3.1.2. We then formalize the problem of (automatic) summarization in Section 3.1.3.

3.1.1 The Purposes of Reading

Before we address summarization, we want to become more specific about the purpose of summarization. According to Mullis, Kennedy, Martin, and Sainsbury (2006), reading can be divided into two areas with two different purposes.

The first purpose of reading is *reading for literary experience*. In reading for literary experience, „the reader becomes involved in imagined events, settings, actions, consequences, characters, atmosphere, feelings, and ideas [...]“ (Mullis et al., 2006). Reading for literary experience is not focused on gathering information but rather on entertainment, relaxation, and self-discovery. Hence, summarization does not apply directly to this type of reading. Someone who wants to enjoy a well-written book, for example, would perhaps refrain from reading a summary of the book instead of the book. Hence, creating summaries for literary experience is not in the focus of this thesis. Note, however, that this does not mean that summarizing books that have been written for literary experience does not make sense. A summary of a book can, for example, satisfy the information need of a person whether or not the person will enjoy reading the book. The purpose of reading the summary is in this example, however, not for literary experience but for gathering information, which is the second purpose of reading according to Mullis et al. (2006).

The second purpose of reading is *reading to acquire and use information* in which *"the reader engages with types of texts where she or he can understand how the world is and has been, and why things work as they do (...)"* (Mullis et al., 2006). This type of reading is directly related to information gathering and summarization. The reader who is reading to acquire and use information is interested in reading the most important information. According to the previously provided information importance definition, the most important information is the information that has the most substantial impact on the reader.

Note, however, that also mixtures of both purposes of reading exist. People might read, for example, a diary about a trip around the world because they enjoy reading the diary and simultaneously want to gather information about other countries. Since we focus in this thesis on information importance estimation, we do not consider reading for literary experience but focus solely on reading to acquire and use information.

3.1.2 Text Summarization Quality Criteria

After specifying that we focus on summaries which are supposed to help people to acquire and use information, we review in this section various definitions of *summarization* that can be found in the scientific literature as well as in other sources. We highlight key terms which are considered relevant for summarization.

Two well-known non-scientific sources define (automatic) summarization as follows. Wikipedia[1] defines automatic summarization as to „*the process of* ***shortening*** *a text document with software, in order to create a summary with the* ***major points*** *of the original document.*" According to the Oxford Dictionaries,[2] to summarize means to „*give a* ***brief*** *statement of the* ***main points*** *of (something)*". Hence, according to both sources a summary contains the main points of the original document. Furthermore, a reduction of length seems to be crucial.

In scientific literature, many more definitions can be found. A well-known definition is given by Mani (2001) where summarization is defined as „*take an information source, extract content from it, and present the* ***most important content*** *to the user in a* ***condensed form*** *and in a manner sensitive to the* ***users or applications needs***". Sparck Jones (1999) defines „*a summary as a* ***reductive*** *transformation of source text to summary text through content reduction by selection and/or generalization on what is* ***important*** *in the source.*" According to Saggion and Lapalme (2002), „*a summary is a* ***condensed*** *version of a source document having a recognizable genre and a very specific purpose: to give the reader an exact and concise idea of the contents of the source.*"

„*A summary can be loosely defined as a text that is produced from one or more texts, that conveys* ***important information*** *in the original text(s), and that is* ***no longer than*** *half of the original text(s) and usually*

[1] https://en.wikipedia.org/w/index.php?title=Automatic_summarization&oldid=822496672

[2] https://en.oxforddictionaries.com/definition/summarize

significantly less than that“, according to Radev, Hovy, and McKeown (2002). Similarly, Hovy (2005) defines a summary as follows: „*A summary is a text that is produced from one or more texts, that contains a significant portion of the information in the original text(s) and that is* ***no longer than*** *half of the original text(s).*“. According to Torres-Moreno (2014), „*an automatic summary is a text generated by a software, that is coherent and contains a significant amount of* ***relevant*** *information from the source text. Its compression rate* τ *is* ***less than a third of the length*** *of the original document.*“

The first three sources mention that a summary is a reduction of the original document. Two sources state that summaries should contain the most important content in the source documents (Mani, 2001; Sparck Jones, 1999) whereas one source specifies that a summary should give an idea of the contents of the source (Saggion & Lapalme, 2002). The user or application needs are mentioned only by one source (Mani, 2001).

The three last sources are more specific about the length of the summary. Two requests that the summary is no longer than 50% of the original document(s) (Hovy, 2005; Radev et al., 2002) whereas the third source requests a maximum length of one third (Torres-Moreno, 2014). Two sources state explicitly that a summary should contain important or relevant information (Radev et al., 2002; Torres-Moreno, 2014).

Cleveland and Cleveland (2013) define an abstract, which might be considered as a special kind of summary, as follows: „*An abstract summarizes the* ***essential contents*** *of a particular knowledge record, and it is a true* ***surrogate*** *of the document*“ According to this definition, an abstract can be used as a replacement of the original document. Length reduction is not explicitly stated but might already be considered by the term „summarizes“.

We summarize the result of the analysis in Table 3.1. ’Important information’ and ’major parts’ are displayed in the same color since they are often used synonymously in automatic summarization. However, with the formal definition of information importance, we see in the following that a summary containing the most important information and a summary containing the main/major parts of the input documents are not necessarily the same.[3]

In basically all sources, it is specified that a summary has to be a reduction of the input documents. Some sources are more specific (i.e., requires a particular amount of reduction), and others are less specific about the resulting length of the summary. Many of the sources also explicitly mention that a summary has to contain the most important information in the source documents. It is, however, only specified by one source (Mani, 2001) that the most important information according to a user or application has to be extracted. Both Wikipedia and Oxford Dictionary state that a summary should contain the major or main points of the document. Note that summarizing the major points of a document refers to user-independent summarization since the main points of a document does not depend on a user but is an intrinsic property of the source document. Since this thesis is concerned with information importance estimation capabilities, and we have specified that the importance of information can only be estimated in the context of a receiver, we are more interested in user-dependent summarization. Other sources (Radev et al., 2002; Sparck Jones, 1999) are not specific on what exactly they refer to by using the term

[3] For an example, see the soccer transcript discussed in Section 4.1.

source	length	important information	major parts	user needs	surrogate
Wikipedia	✓		✓		
Oxford Dictionaries	✓		✓		
Cleveland and Cleveland (2013)	(✓)	✓			✓
Sparck Jones (1999)	✓	✓			
Radev, Hovy, and McKeown (2002)	✓	✓			
Hovy (2005)	✓				
Torres-Moreno (2014)	✓	?	?		
Saggion and Lapalme (2002)	✓				
Mani (2001)	✓	?	?	✓	

Table 3.1.: Overview of relevant aspects for good summaries. 'Important information' and 'major parts' are displayed in the same color since they are often used synonymously in automatic summarization.

importance. Hence, it is not clear whether they refer to user-specific importance like Mani (2001) or the document-specific main points of a document.

We conclude that summarization is specified in the literature as a process that reduces the length of an input document (or multiple input documents). After the reduction, the content of the summary should either contain the main parts of the input document(s) or the most important content for a user or application need. We focus on this thesis on the second aspect and want to create summaries that contain the most important information for a particular user or user group. We continue in the next section with a formal definition of summarization.

3.1.3 A Problem Definition for Summarization

After the informal definitions discussed in the previous, we now provide a formal definition of summarization. To this end, we specify what we mean with terms such as *medium*, *information content*, and summary.

Definition 9 (Medium). A **medium** $\mathfrak{m}$ (or information carrier) is a communication means used to store and deliver information. The set of all media of a particular type (medium space) is denoted by $\mathfrak{M}_t$ where t denotes the type of the media. $\mathfrak{M}_{text}$, for example, denotes all texts and $\mathfrak{M}_{image}$ denotes all images. The type of the medium space can be omitted for the sake of brevity.

Examples of different types of media are texts, images, videos, and maps. The corresponding media spaces are the set of all text, images, videos, and maps, respectively.

Definition 10 (Size of a Medium). Given a medium space $\mathfrak{M}$, we denote the **size** (or length) of the medium by function $|.| : \mathfrak{M} \rightarrow [0, \infty)$. There are multiple ways to measure the length of a given medium. The size of texts, for example, can be measured in number of words or number of characters. The size of images can, for example, be measured in number of pixels.

Definition 11 (Information Pieces). Information pieces (or information nuggets) are atomic, independent elements describing the state of a system. We denote the set containing all information nuggets (information space) by $\mathfrak{I}$.

Definition 11 specifies information pieces as *atomic* and *independent*. With atomic, we refer to the property that information pieces cannot be divided into smaller elements. Information pieces are furthermore independent, which means that they do not overlap. Both assumptions allow a simpler formulation of the remaining statements and have also been considered by prior work (Nenkova & Passonneau, 2004).

Definition 12 (Information Content). Let $\mathfrak{M}$ be a medium space and $\mathfrak{I}$ the set of all information nuggets. Let function $\mathfrak{c} : \mathfrak{M} \rightarrow \mathscr{P}(\mathfrak{I})$ extract all information nuggets which are contained in a medium. The output of $\mathfrak{c}$ is called **information content**.

Note that the output of $\mathfrak{c}$ is a set which means that all information nuggets contained in $\mathfrak{M}$ are counted only once. For simplicity, we also use the definition $\mathfrak{c} : \mathfrak{M} \rightarrow \mathfrak{I}$ which outputs the information content of a set of media.

Definition 13 (Summary). Given a medium space $\mathfrak{M}_t$, size function $|.|$, and a set of media $\{X_1, \ldots, X_n\} \subset \mathfrak{M}_t$, a medium $Y \in \mathfrak{M}_t$ with

1. $|Y| < \sum_{i=1}^{n} |X_i|$
2. $\mathfrak{c}(Y) \subset \bigcup_{i=1}^{n} \mathfrak{c}(X_i)$

is called a **summary** of set $\{X_1, \ldots, X_n\}$. A summary is called **single-document summary** if $n = 1$ and **multi-document summary** if $n > 1$.

Condition 1 in Definition 13 (i.e., $|Y| < \sum_{i=1}^{n} |X_i|$) requires that the size of the summary is smaller than the size of the input. This property is also frequently reported in the literature, as discussed above. Furthermore, it is one of the two key properties which makes summarization an interesting task for information importance estimation since reducing the size of a medium is related to selecting information nuggets according to a utility.

Condition 2 requires that the summary only contains information which is also contained in the input medium. Hence, there cannot be new information in a summary that has not already been included in the original input documents. Without this requirement, a summary could contain any content without being related to the source documents. In this case, summarization would not be a reduction of information content.

According to Definition 13, both input media and output medium have to be in the same medium space $\mathfrak{M}_t$. An image, for example, cannot be summarized by a text according to this definition. This restriction is important to be able to define a meaningful size function.

Figure 3.1 illustrates single-document summaries for three different media. We denote the input medium on the left by $\mathfrak{m}_{\text{input}}$ and the output medium (i.e., the summary) on the right by $\mathfrak{m}_{\text{output}}$. Figure 3.1a illustrates a long text[4] on the left ($|\mathfrak{m}_{\text{input}}| = 62$ words) which is summarized by a short text on the right ($|\mathfrak{m}_{\text{output}}| = 13$ words). Details such as the exact date of the U.S. presidential election are omitted in the summary and no additional information is added to the summary. Figure 3.1b shows a graph on the left which illustrates large cities around Darmstadt. The original graph consists of 9 nodes. The summary graph on the right only consists of 3 nodes. Smaller cities are not displayed in the summary graph. The original image[5] on the left in Figure 3.1c has a size of 512x512 pixels. The summary on the right is a smaller version of the original image and consists of only 256x256 pixels. Details of the original image are not very well visible anymore. The main content of the image is, however, still visible. We focus in this thesis on summaries of textual document such as displayed in the first summarization example.

Definition 14 (Summarizer). A function $\mathfrak{S} : \mathscr{P}(\mathfrak{M}_t) \rightarrow \mathfrak{M}_t$ such that $\mathfrak{S}(\mathrm{I}) = \mathrm{O}$ is a summary of $\mathrm{I}, \forall \mathrm{I} \subset \mathscr{P}(\mathfrak{M}_t)$ is called a t-**summarizer** (or summarization system). We consider in this thesis only textual summaries. Due to simplicity, we will omit information about t and will always refer to texts if not stated otherwise. Summarizers can, for example, be humans, as well as computer systems.

Research in *automatic summarization* aims at building automatic (i.e., non-human) summarizers. We formalize in the next section what *optimal* summaries are.

3.2 Optimal Summaries

We discuss in this section how optimal summaries can be defined. An optimal summary contains the most important information (Section 3.1.2). Since we have already specified what we mean with *important information*, we are able to formally define how the best possible summary of a document or a document collection looks like. A key property is the audience, which is at the center of the provided importance definition. Hence, we can only define optimal summaries with the audience in mind. We start with defining how the optimal summary for a single person looks like and generalize the definition to arbitrarily sized sets of people. We explain how this theoretical discussion relates to a practical implementation of a well-known evaluation method called Pyramid method (Nenkova & Passonneau, 2004).

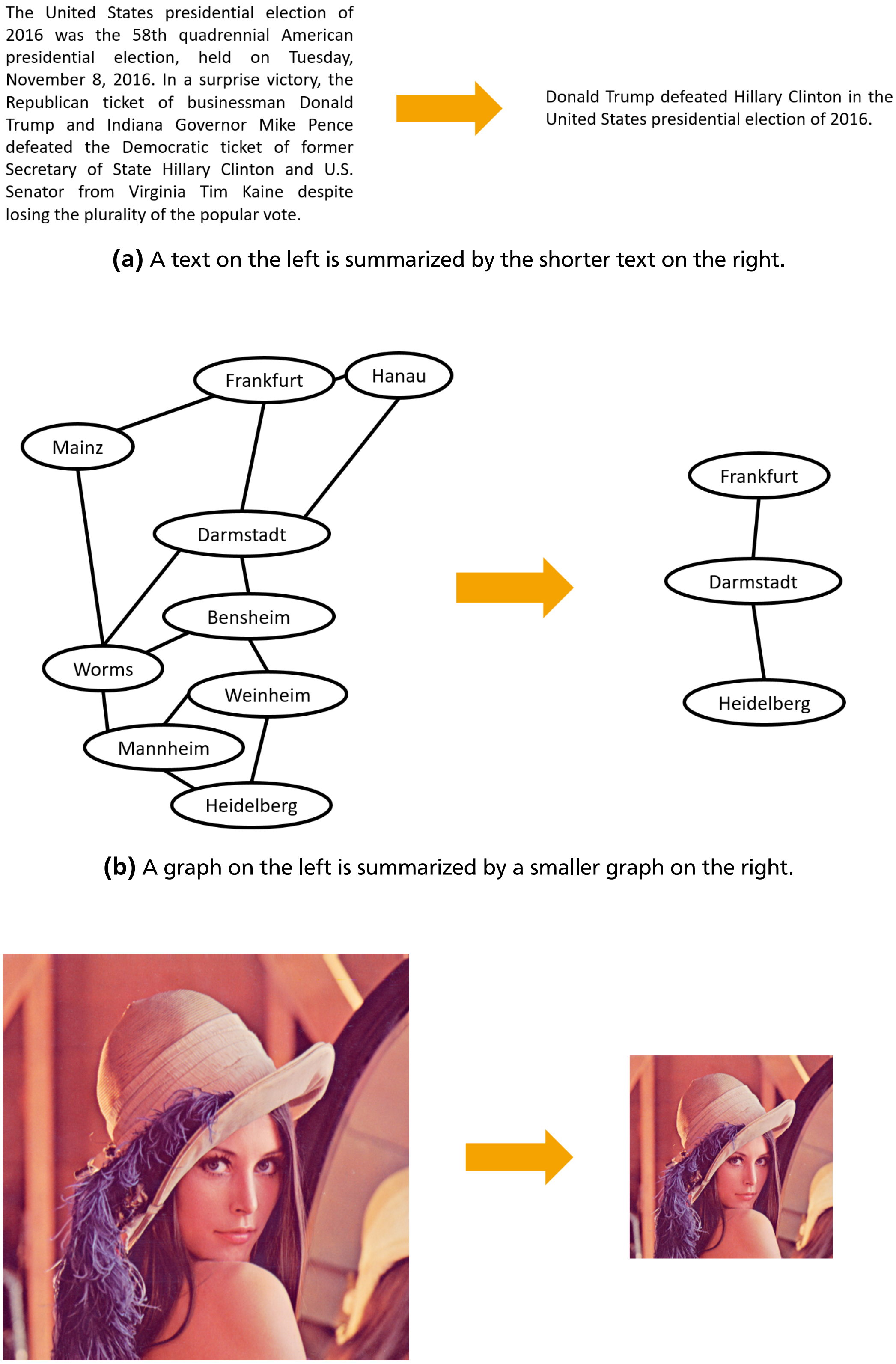
In Section 3.2.1, we define the optimal summary for one reader based on the definition of importance given in the previous chapter. In Section 3.2.2, we generalize this definition to larger audiences.

3.2.1 The Optimal Personal Summary

We now combine the definition of information importance provided in Chapter 2 with the problem of summarization defined in Section 3.1.3. Since we focus on summaries, which contain the most important information to a user, we can use both definitions to deduct how the optimal personal summary can be defined.

[4] Source: https://en.wikipedia.org/w/index.php?title=United_States_presidential_election,_2016&oldid=834116413

[5] Source: https://en.wikipedia.org/w/index.php?title=Lenna&oldid=830125071



The United States presidential election of 2016 was the 58th quadrennial American presidential election, held on Tuesday, November 8, 2016. In a surprise victory, the Republican ticket of businessman Donald Trump and Indiana Governor Mike Pence defeated the Democratic ticket of former Secretary of State Hillary Clinton and U.S. Senator from Virginia Tim Kaine despite losing the plurality of the popular vote.

Donald Trump defeated Hillary Clinton in the United States presidential election of 2016.

(a) A text on the left is summarized by the shorter text on the right.

(b) A graph on the left is summarized by a smaller graph on the right.

(c) A picture on the left is summarized by a smaller picture on the right.

Figure 3.1.: Illustration of summarization examples for different media types. The input medium on the left is in every example larger (in terms of number of words in Figure 3.1a, number of nodes in Figure 3.1b, and number of pixes in Figure 3.1c) than the summary on the right.

Based on the information content function $\mathfrak{c}$ in Definition 12 and the personal importance function $\text{Imp}_p(\mathfrak{i})$ in Definition 8, we can define the utility function of a text document (e.g., for a summary) with respect to a particular person p.

Definition 15 (Utility of a Text). The **Utility of a text document** D with respect to person p with importance function Imp_p is defined as $\bar{u}(D) = \sum_{\mathfrak{i} \in \mathfrak{c}(D)} \text{Imp}_p(\mathfrak{i})$.[6]

Hence, the utility of a text document equals the sum of the importance scores of the contained information nuggets. The more important information nuggets a text contains, the more important it is. Note that including redundant information multiple times to a text does not increase its utility since the function $\mathfrak{c}$ maps to a set (see Definition 12). Each information nugget is therefore only contained at most once. Creating a text with a maximum utility is simple in theory by just concatenating all available text. This solution is, however, impractical since no human is able to read such a large amount of text. We also motivated this thesis with the goal to reduce the reading effort of humans while retaining a good information consumption. This aim is the reason why automatic summarization tries to reduce the amount of text while retaining the most important information. Therefore, we include a length constraint in the further discussion in which we define the optimal personal summary and the optimal summary for multiple recipients.

Definition 16 (Optimal Personal Summary). Let $\mathfrak{T}$ be a set containing all possible text documents, $X_i \subset \mathscr{P}(\mathfrak{T})$ a set of input documents, and p a person. The **optimal personal summary** of X_i for p with maximal length l is the text $Y^\star$ with

1. $|Y^\star| \leq l$,
2. $\mathfrak{c}(Y^\star) \subseteq \mathfrak{c}(X_i)$, and
3. $\bar{u}(Y^\star) \geq \bar{u}(Y) \forall Y \in \mathfrak{T}, |Y| \leq l$.

Hence, we request three properties. The first property $|Y^\star| \leq l$ ensures that the summary is not longer than intended. Without a length constraint, it would be simple to create arbitrarily long texts with a very high utility. $\mathfrak{c}(Y^\star) \subseteq \mathfrak{c}(X_i)$ ensures that all information nuggets in the summary are also contained in the source documents (i.e., the summary is actually a reduction of the input documents (Mani, 2001; Sparck Jones, 1999)). $\bar{u}(Y^\star) \geq \bar{u}(Y), \forall Y \in \mathfrak{T}$ formalizes the desired property that there is no summary with a higher utility score than the optimal summary. We did not specify which summary is optimal in the case where multiple optimal summaries exist. In this case, we choose the shortest summary with the maximal utility as the optimal summary.

The definition of optimal personal summary implicitly rewards summaries that do not contain redundant information. Including redundant information consumes space which could be used to include a new

[6] We omit in the definition of $\bar{u}$ that the function is parameterized by person p.

non-redundant information nugget which is rewarded since the utility of a summary increases if space which is used for redundant information is used to include non-redundant information.

Even though it is best to create a summary individually for every user, it might be in practice challenging to produce user-specific summaries. In newspapers, for example, it is not possible to exactly match the individual needs of every reader since there is only one article for a potentially huge set of readers. A good journalist anticipates the information needs of the whole target audience rather than one single person. A journalist working for a local newspaper in Darmstadt, for example, would perhaps not report about local events in another city which are not relevant to the readers in Darmstadt. Hence, we are interested in how the optimal summary for a broader target audience can be defined.

3.2.2 Generalization to Larger Target Audiences

Based on the previous definition for a personal summary, we can generalize to an optimal summary for a larger target audience.

Definition 17 (Optimal Summary). Given a set of input documents $D_i \subset \mathscr{P}(\mathfrak{T})$, the **optimal summary** for a set of people $P = \{p_1, \ldots, p_n\}$ equals to the optimal personal summary of the prototypical person $\hat{p}$ with $\mathrm{Imp}_{\hat{p}}(\mathfrak{i}) = \frac{1}{n} \cdot \sum_{j=1}^{n} \mathrm{Imp}_{p_j}(\mathfrak{i}), \mathfrak{i} \in \mathfrak{I}$.

It can be observed that the definition of the personal summary given in Definition 16 is a special case (with $n = 1$) of the definition given in Definition 17. Hence, Definition 17 is a generalization of Definition 16 and can be used instead of the optimal personal summary.

The optimal group summary maximizes the total average reward given a population of readers by generating the optimal summary for $\hat{p}$. The generated summary does not have to be fair in the sense that the information need of every group member is at least somehow reflected in the summary. If one person has different information needs with respect to the prototypical person in the group, he or she might not be satisfied with the resulting summary. The definition of $\mathrm{Imp}_{\hat{p}}(\mathfrak{i})$ can be modified to consider other properties such as fairness. For example, a fair summary has to reflect the information need of every individual at least to a certain extent.

Connection to the Pyramid Method

The previously discussed concepts of optimal summaries are rather abstract, mainly because it is not possible in practice to estimate the importance of information nuggets for individual people. There exist, however, practical evaluation methods that can be viewed as concrete implementations of the previously defined optimal summaries.

Very close to the discussed and rather abstract definitions is the Pyramid method (Nenkova & Passonneau, 2004). The key idea of the Pyramid method is to specify a set of information nuggets called summary content units (SCUs), to assign weights to the SCUs, and to define the quality of summaries as the sum of the weights of the contained SCUs. Each SCU represents a piece of particular information which is contained in the documents. Hence, SCUs can be viewed as concrete instances of the abstract idea of information nuggets $\mathfrak{i} \in \mathfrak{I}$. The function $\mathfrak{c}$ which extracts all information nuggets from a text can, therefore, be interpreted as an abstract version of the concrete function which extracts all SCUs from the text documents. Another frequently used approximation of information nuggets are bigrams (Gillick, Favre, & Hakkani-Tür, 2008; Zopf, Loza Mencía, & Fürnkranz, 2016a). The weight of the SCUs is defined by analyzing reference summaries written by humans. If a person includes an SCU in his or her summary, the weight of the respective SCU is increased. This weighting scheme can be viewed as concrete implementation of the importance function Imp. An SCU is considered to be important to a particular person p if he or she includes the SCU in his or her reference summary. Different to the definition of Imp, an SCU is either important or not important for person p since it is either included or not included in the reference summary provided by p. The utility of automatically generated summaries is computed by summing the weights (either 0 or 1 if only one reference summary is given) of the SCUs which are included in the summary. Hence, the more SCUs are considered important according to the reference summary of person p, the higher the utility of the automatically generated summary is. The optimal summary, in this case, contains all the SCUs which are considered important by person p, which matches the definition of the optimal personal summary very well. If multiple references are available, which is usually the case when the Pyramid method is applied, the weight of an SCU equals to the sum of every individual weight provided by every person for this particular SCU. Hence, the optimal summary for a group of people is the summary which includes the most valuable SCUs according to the Pyramid method. This weighting scheme aligns with the proposed creation of an average person $\hat{p}$, which is defined as the average of the information need of all people in the target audience. Note that the target audience equals to the set of people who created reference summaries. Furthermore, the Pyramid method proposes a definition of optimality which ”*ignores many complicating factors (e.g., ordering, SCU interdependency)*” (Nenkova & Passonneau, 2004). We also make this simplification in our model.

3.3 Tackling Automatic Summarization with Extractive Summarization

In this section, we focus on a particular approach to summarization called *extractive summarization*. As defined in Section 3.3.1, an extractive summarizer splits the input documents into parts (usually sentences), selects a subset of the parts, and concatenates the parts to form a summary. In Section 3.3.2, we explain how greedy algorithms can be used for extractive summarization.

3.3.1 Definition of Extractive Summarization

We extend Definition 13 to define the term *extractive summary*.

Definition 18 (Extractive Summary). Given a medium space $\mathfrak{M}_t$, size function $|.|$, a set of media $\{X_1, \ldots, X_n\} \subset \mathscr{P}(\mathfrak{M}_t)$, a partition function $\mathfrak{p} : \mathfrak{M}_t \to \mathfrak{M}_t$ which partitions $\mathfrak{M}_t$ in to smaller parts and a composition function $\mathfrak{q} : \mathfrak{M}_t \to \mathfrak{M}_t$ which builds a summary by composing smaller parts, a medium $Y \in \mathfrak{M}_t$ with

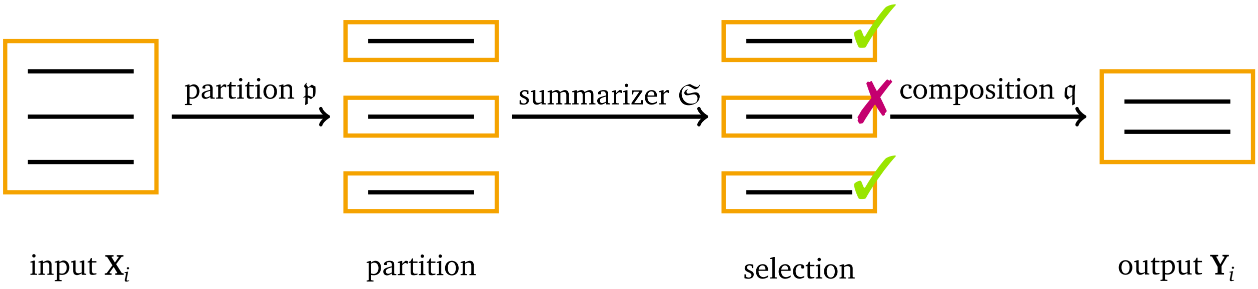


Figure 3.2.: Extractive summarization starts input document(s) on the left. A partition function $\mathfrak{p}$ partitions the input document(s) in to a set of parts (usually individual sentences). The extractive summarizer $\mathfrak{S}$ selects a subset of the parts. A composition function $\mathfrak{q}$ creates the output (i.e., the summary) by composing the selected parts. The composition function is usually implemented by a simple concatenation (with or without a sentence reordering).

1. $|Y| < \sum_{i=1}^{n} |X_i|$
2. $\mathfrak{c}(Y) \subset \bigcup_{i=1}^{n} \mathfrak{c}(X_i)$
3. $Y = \mathfrak{q}(\mathfrak{p}(X_i))$

is called an **extractive summary** of the media set $\{X_1, \ldots, X_n\} \subset \mathscr{P}(\mathfrak{M}_t)$.

Condition 1 and 2 in Definition 18 for extractive summaries are the same as in Definition 13. Condition 3 states that the output Y has to be created by composing parts produced by the partition function $\mathfrak{p}$. Extractive summarization extracts a subset of sentences from a given set of sentences while simultaneously maximizing a utility function and is, therefore, a *optimal subset selection problem*.

In text-based extractive summarization, $\mathfrak{p}$ usually maps from text documents to individual sentences. Sentences splitters such as The Stanford Tokenizer[7] (Manning et al., 2014) or DKPro[8] (Eckart de Castilho & Gurevych, 2014) are therefore concrete implementations of function $\mathfrak{p}$. The composition function $\mathfrak{p}$ is usually implemented by just concatenating the selected sentences. To create a summary of a video, a reasonable choice for $\mathfrak{p}$ would be the partition of the video into individual pictures or short video sequences. The composition function $\mathfrak{q}$ would then be a concatenation of individual pictures or short video sequences to generate a full movie summary.

Definition 19 (Extractive Summarizer). Summarizers $\mathfrak{S}$ which only create extractive summaries according to Definition 18 are called **extractive summarizers.**

Extractive summarization is visualized in Figure 3.2.

Focusing on extractive summarization is appealing for the research questions covered by this thesis since it requires information importance estimation capabilities. We present in the following a solution approach to extractive summarization called greedy sentence selection.

[7] https://nlp.stanford.edu/software/tokenizer.shtml
[8] https://dkpro.github.io/

As described above, extractive summarization is an optimal subset selection problem. Optimal subset selection problems can be viewed as search problems. The search space is the space of all possible subsets (i.e., the powerset $\mathscr{P}(\mathfrak{p}(X_i))$) of the set produced by $\mathfrak{p}$ for input documents X_i. Selecting a subset of sentences is equivalent to maximum set coverage and is, therefore, an np-hard problem (Filatova & Hatzivassiloglou, 2004; Hochbaum, 1997). Many optimization algorithms exist which find optimal or approximative solutions. In the following, we focus on iterative greedy selection approaches.

Let I (I for **I**nput) be a list of sentences with length n. Let $\mathbf{R} : \mathbb{N} \rightarrow I$ be a ranking function which maps from natural numbers (i.e., from a position) to the corresponding sentence in the ranking. Hence, $\mathbf{R}(i)$ returns the sentence at position i in the ranking. Algorithm 3 presents the greedy sentence selection algorithm which generates a subset $O \subseteq I$ (O for **O**utput) until a given length criterion $||O||$ is not satisfied anymore.

Algorithm 3 Greedy Sentence Selection

```
list of all input sentences $I = \mathbf{s}_1, \ldots, \mathbf{s}_n$
$\mathbf{R}$ = ranking of $I$
desired summary length $l$
$O \leftarrow \emptyset, i \leftarrow 1$
while $||O|| < l$ and $i < n$ do
    $O \leftarrow O \cup \mathbf{R}(i)$
    $i \leftarrow i + 1$
end while
return $O$
```

The algorithm is said to select sentences greedily because it iteratively removes always the highest ranked sentence without considering future consequences and without considering to revise already made decisions. The major advantage of the greedy algorithm is its simplicity, which allows a quick execution in comparison to more complex search algorithms. Due to the simplicity of the algorithm, the time complexity of greedy algorithms is usually dominated by finding a suitable ranking **R**, which is required as input for the algorithm.

The algorithm returns the optimal solution if the problem has a matroid structure (Papadimitriou, 1981) and gives a constant factor approximation to problems with submodular structure (Nemhauser, Wolsey, & Fisher, 1978). It has been shown that extractive summarization with ROUGE (C.-Y. Lin, 2004) as evaluation function (see Section 7.2.3 for details) has a submodular structure (H. Lin & Bilmes, 2011). It is therefore guaranteed that a greedy algorithm achieves at least a ROUGE score of $1 - \frac{1}{e} \cdot \text{ROUGE}_{\text{opt}}$ where $\text{ROUGE}_{\text{opt}}$ is the ROUGE score of the optimal solution, $e = \sum_{i=0}^{\infty} \frac{1}{i!}$ (better known as Euler's number) and the true ROUGE scores of the individual sentences are known.

In classical optimization, the true rewards (or costs, in case of minimization) are known. Examples are the traveling salesman problem and the knapsack problem in which the true distances between nodes and the true rewards and weights of each element is known. In summarization, however, the true ROUGE scores of sentences, for example, are not known and can only be approximated. Hence, no algorithm can guarantee to find the optimal solution according to the true scores. The performance of all algorithms strongly depends on the estimation quality of sentence utilities. For the greedy search, exact sentence utilities are not required since only the rank of each sentence matters.

The algorithms are called greedy since they pick at each decision the locally best action without being concerned about the overall best solution. These kinds of algorithms are appealing since they run in linear time with respect to the number of sentences in the input documents given that the individual rewards for each decision (in summarization usually the sentence utilities) are known. The drawback of greedy methods is that they are not guaranteed to find the best solution. We argue that this drawback is negligible since the correct rewards for the individual choices are not known but can only be approximated. Hence, no algorithm can provide guarantees in this setup.

Greedy Sentence Selection with Redundancy Avoidance

Finding a ranking such that the greedy selection performs well is difficult since the selected sentences do not contribute independently to the utility of summaries. For example, redundancy and synergy effects have to be considered during the generation of the ranking. Redundancy causes the utility of two sentences to be lower than the sum of the individual sentence utilities (i.e., $\dot{u}(\mathbf{s_i} \cup \mathbf{s_j}) < \dot{u}(\mathbf{s_i}) + \dot{u}(\mathbf{s_j})$). This effect can occur in summarization when two sentences with overlapping information are added to the summary. Synergy refers to the effect when the utility of two sentence is higher then the sum of the individual sentence utilities (i.e., $\dot{u}(\mathbf{s_i} \cup \mathbf{s_j}) > \dot{u}(\mathbf{s_i}) + \dot{u}(\mathbf{s_j})$). This effect occurs in summarization, for example, if sentence $\mathbf{s_j}$ contains very important information which cannot be understood (e.g., because of a missing pronoun antecedent (Paice, 1990)) without the context of another sentence $\mathbf{s_i}$. The problem of considering redundancy during the learning of $\mathbf{R}$ can be mitigated by modifying the greedy selection algorithm.

Algorithm 4 Greedy sentence selection with redundancy avoidance

```
list of all input sentences $I = \mathbf{s}_1, \ldots, \mathbf{s}_n$
$\mathbf{R}$ = ranking of $I$
desired summary length $l$
similarity function sim
similarity threshold $\theta$
$O \leftarrow \emptyset, i \leftarrow 1$
while $||O|| < l$ and $i < n$ do
    if $\mathrm{sim}(\mathbf{R}(i), O) < \theta$ then
        $O \leftarrow O \cup \mathbf{R}(i)$
    end if
    $i \leftarrow i + 1$
end while
return $O$
```

Algorithm 4 shows a greedy algorithm that greedily selects the highest-ranked sentence only if it is not too similar to previously selected sentences in O. If the highest-ranked sentence exceeds a similarity threshold θ it is skipped, and the algorithm continues with the next sentence in the ranking. Note that the time complexity of the greedy algorithm can increase substantially if the computation of $\mathrm{sim}(\mathbf{s}_{\mathbf{R}(i)}, O)$ is expensive. Sometimes, additional cut-off criteria are added in addition to the redundancy avoidance. Sentences with a length below a particular fixed threshold, for example, are sometimes skipped as well even if they are not similar to sentences in the output.

Greedy Sentence Selection with Re-ranking

An alternative to a modification of the greedy selection process in order to accommodate for redundancy is a modification of the permutation $\mathbf{R}$. Algorithm 5 is different from Algorithm 4 because no threshold is used as hard constrained to decide whether or not a sentence has to be skipped. Instead, a weighted average of a sentence utility $\dot{u}$ and sentence redundancy is computed such that a sentence $\mathbf{s_i}$ with a very high utility can be included into the summary O even if the summary already contains a sentence $\mathbf{s_j}$ which is similar to $\mathbf{s_i}$. How much the redundancy has to be considered is specified by parameter λ. The redundancy is computed in Algorithm 5 with similarity function $\mathrm{sim}(\mathbf{s_i}, O)$. After a sentence has been added to summary O, a re-computation of $\mathbf{R}$ (re-ranking) has to be performed since O changed and the ranking function depends on O. Hence, the ranking might change as well.

Algorithm 5 Greedy sentence selection with re-ranking

```
list of all input sentences $I = \mathbf{s_1}, \ldots, \mathbf{s_n}$
desired summary length $l$
similarity function sim
$\widetilde{\dot{u}(\mathbf{s_i})} = \lambda \cdot \dot{u}(\mathbf{s_i}) - (1 - \lambda) \cdot \mathrm{sim}(\mathbf{s_i}, O)$
$\mathbf{R}$ = ranking of $I$ subject to $\widetilde{\dot{u}(\mathbf{R}(1))} \geq \cdots \geq \widetilde{\dot{u}(\mathbf{R}(n))}$
$O \leftarrow \emptyset, i \leftarrow 1$
while $\|O\| < l$ and $i < n$ do
    $O \leftarrow O \cup \mathbf{R}(i)$
    $i \leftarrow i + 1$
    recompute $\mathbf{R}$ subject to $\widetilde{\dot{u}(\mathbf{R}(1))} \geq \cdots \geq \widetilde{\dot{u}(\mathbf{R}(n))}$
end while
return $O$
```

Maximal Marginal Relevance

A well-known special case of the previously described modification is the maximal marginal relevance re-ranking (Carbonell & Goldstein, 1998). The utility of sentence $\mathbf{s_i}$ is computed with a second similarity function which computes the similarity between $\mathbf{s_i}$ and the input I. Hence, a sentence has to be similar to the input and dissimilar to the (intermediate) output O to be ranked highly. Algorithm 6 lists a greedy selection algorithm with maximal marginal relevance re-ranking. The similarity function which is used to compute the similarity between $\mathbf{s_i}$ and the input is called $\mathrm{sim}_{\mathrm{in}}$ and the similarity function which is used to compute the similarity between $\mathbf{s_i}$ and the output is called $\mathrm{sim}_{\mathrm{out}}$. In a simple setup, both similarity functions can be instantiated with the same similarity function.

Algorithm 6 Greedy sentence selection with maximal marginal relevance re-ranking

```
list of all input sentences $I = \mathbf{s_1}, \ldots, \mathbf{s_n}$
desired summary length $l$
similarity functions $\text{sim}_{\text{in}}, \text{sim}_{\text{out}}$
$\overparen{\dot{u}(\mathbf{s_i})} = \lambda \cdot \text{sim}_{\text{in}}(\mathbf{s_i}, I) - (1 - \lambda) \cdot \text{sim}_{\text{out}}(\mathbf{s_i}, O)$
$\mathbf{R}$ = ranking of $I$ subject to $\overparen{\dot{u}(\mathbf{R}(1))} \geq \cdots \geq \overparen{\dot{u}(\mathbf{R}(n))}$
$O \leftarrow \emptyset, i \leftarrow 1$
while $\|O\| < l$ and $i < n$ do
    $O \leftarrow O \cup \mathbf{R}(i)$
    $i \leftarrow i + 1$
    recompute $\mathbf{R}$ subject to $\overparen{\dot{u}(\mathbf{R}(1))} \geq \cdots \geq \overparen{\dot{u}(\mathbf{R}(n))}$
end while
return $O$
```

3.4 Shortcomings of Automatic Summarization for Information Importance Estimation

We discuss in the following several shortcomings of using text summarization to estimate the information importance estimation capabilities of systems.

Comparing Semantic Content of Texts is Difficult

To evaluate performance in summarization properly, it is necessary to estimate the (semantic) similarity between generated summaries and reference summaries properly. The method of choice today is a text similarity estimation method called ROUGE (C.-Y. Lin, 2004). The idea of ROUGE is to estimate the semantic similarity by computing the overlap of n-grams between an automatically generated summary and one or more reference summaries.[9] Due to lexical ambiguity, vagueness, and uncertainty, which occur in natural language, the quality of semantic similarity estimation for two texts based on lexical comparison is fundamentally limited. Hence, ROUGE is also not able to reliably estimate the semantic similarity of texts, which has been criticized many times in the research community.

Summarization is not Independent of Redundancy

A second issue of using summarization for information importance estimation capabilities is the requirement to handle information importance and redundancy jointly since good summaries are expected not to contain the same information multiple times. Hence, it cannot be evaluated in summarization if a sentence has not been included in the summaries because it has been considered to be unimportant or redundant. The information importance estimation abilities can, therefore, not be assessed in isolation. In a good experiment, however, it is desirable to reduce the number of disturbing factors which might influence the result of the experiment.

[9] More details about ROUGE can be found in Section 7.2.3

A Summary is a Set of Information

Furthermore, extractive summaries are only sets of information and do not indicate which information in the set is more or less important for the summary. It is a coarse-grained binary classification into important and not important (also including redundancy avoidance). Hence, summarization systems do not have to have a fine-grained estimation of information importance.

Linguistic Quality

Furthermore, good summaries are expected to have good linguistic quality, which includes factors such as a good discourse structure and the absence of spelling mistakes. The ability to achieve high linguistic quality is, however, not directly related to the ability to estimate information importance. Similarly to redundancy avoidance, it would be best to remove as many disturbing factors as possible from the evaluation.

Summaries Have a Length Restriction

Summarization is usually performed with regard to a length restriction. The length parameter adds additional complexity to the evaluation and makes interpretation of results harder. A summarization system might be good at producing short summaries while having difficulties to produce longer summaries.

3.5 Alternative Evaluation Strategies

We introduced automatic summarization in Section 3.1 as a means to evaluate the information importance estimation capabilities of (automatic) systems because automatic summarization is the most prominent research area which is related to this task. Evaluating information importance estimation capabilities with summarization has, however, also some disadvantages which have been discussed at the end of the last section. We propose three new ways to test information importance estimation capabilities in Section 3.5.1, Section 3.5.2, and Section 3.5.3 which do not suffer from the discussed disadvantages.

3.5.1 Predicting Utility Scores of Information Nuggets

The first non-summarization method proposed in this section is to estimate the ability of systems to predict utility scores of information nuggets correctly. Unfortunately, it is difficult to find reliable and sound utility scores for information nuggets. In practice, it is, for example, not possible to find correct values for the importance of information as described in Section 2. Approximations of true importance scores can, however, be generated easily. ROUGE scores of individual sentences can, for example, be used as target values. Note that this evaluation should not be confused with summarization based on sentence regression (see Section 6.4 for details). The crucial difference is that sentence regression systems are evaluated based on the quality of the resulting summary. Instead, we propose to evaluate systems based on the predicted utility scores. The new approach allows avoiding the disadvantages of summarization, such as redundancy avoidance and a coarse-grained sentence importance prediction.

3.5.2 Evaluation of Sentence Rankings

The second evaluation method is based on rankings of information. Given the three sentences A, B, and C in the introduction in Chapter 1, a ranking of all information nuggets according to their importance might be: $B \succ C \succ A$. Using rankings instead of utility prediction eliminates the need for 'correct' utilities, which makes ranking more appropriate for tasks in which it is difficult to find good approximations of the utilities.

We propose to use two rank correlation metrics to estimate the quality of the predicted ranking. The Kendall rank correlation coefficient (Kendall's tau) Kendall (1938) computes the number of concordant pairs in the rankings. All disagreements are equally weighted, i.e., ranking mistakes in the bottom and the top part of the ranking are equally penalized by Kendall's tau.

Having a good agreement in the top part of the ranking might be considered to be more important than a good agreement in the bottom part of the ranking. In summarization, for example, it is important to have the most important information ranked highly, whereas the ranking in the lower part of the ranking is not so important. Hence we define a variant of the discounted cumulative gain (DCG) (Järvelin & Kekäläinen, 2002) which is frequently used in information retrieval. We define the *discounted cumulative ranking score* (DCRS) between two ranking functions $\mathbf{R}$ and $\tilde{\mathbf{R}}$ as

$$DCRS(\mathbf{R}, \tilde{\mathbf{R}}) = \sum_{i=1}^{n} \frac{\frac{1}{\mathbf{R}^{-1}(o_i)}}{\ln(\tilde{\mathbf{R}}^{-1}(o_i) + 1)}, \tag{3.1}$$

where $\mathbf{R}^{-1}(o_i)$ and $\tilde{\mathbf{R}}^{-1}(o_i)$ indicate the rank of o_i according to $\mathbf{R}$ and $\tilde{\mathbf{R}}$, respectively. The difference to DCG is that the gain we use $(\frac{1}{\mathbf{R}_f(o_i)})$ is only based on the rank and does not consider the utility of the elements. This is appealing if a ranking has to be compared to an ordinarily scaled list in which no true utilities for the individual elements are known. The example in the introduction is such a case. It is reasonable to say that $B \succ C \succ A$ holds. It is, however, basically impossible to give a good estimate of the individual utilities of the sentences. It is, for example, hard to tell whether B should have a utility of 0.80 or 0.60. Following the most common variant of DCG, we also use a logarithmic discount factor (Y. Wang et al., 2013) and propose to use a normalized version of DCRS (nDCRS) which maps all DCRS scores into $[0, 1]$ (Chen, Liu, Lan, Ma, & Li, 2009). A random permutation of the list yields an nDCRS score of 0.50.

Evaluation with rankings is appealing for multiple reasons. First, no texts written in natural language have to be evaluated or compared, which makes the evaluation much more reliable and sound. Redundancy does not have to be modeled in a ranking. Ranking sentences with similar content to similar positions (e.g., the most important information to the topmost positions) is the expected result and therefore perfectly fine. Hence, evaluating with rankings removes the unclarity introduced by the need for redundancy avoidance in summarization. A ranking is also a fine-grained estimation of information importance compared to the coarse-grained classification in summarization. Systems are not tested whether or not they are able to distinguish important from unimportant sentences, but whether they are able to estimate the relative importance of information. Furthermore, all sentences in a given set have to be ranked, which means that the systems have to express their beliefs about the importance of all sentences. Last but not least, the evaluation of linguistic quality is separated from the evaluation of information importance estimation capabilities.

To use ranking for evaluation, reference rankings have to be available, which is not the case in standard summarization datasets. Standard datasets usually contain reference summaries as expected output. Hence, reference rankings have to be produced first. Fortunately, reference rankings can be generated based on available summarization datasets. Sentences can, for example, be ranked according to the highest similarity with sentences in the reference summary. The task of an information importance estimation system would then be to achieve a high correlation or a high nDCRS without knowing the reference summaries. Another option is to create reference rankings manually. Crowdsourcing may be used, for example, to generate many reference rankings cheaply. The advantage over creating summaries is that crowdworkers, for example, can easily rank three sentences, whereas the construction of summaries is a complex task which is rather not suited for crowdsourcing.

3.5.3 Pairwise Preference Prediction

A special case of ranking construction is pairwise preference prediction. Here, the task is to predict which sentence contains more important information for two given sentences. The accuracy of the prediction can be calculated by counting the number of agreements with a set of given preference labels according to Equation 3.2. Given n pairs of sentences (d_i, r_i), we define the accuracy of a systems's preference prediction $\tilde{\mathbf{R}}$ ability as

$$acc = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(\tilde{\mathbf{R}}^{-1}(r_i) \succ \tilde{\mathbf{R}}^{-1}(d_i)) \tag{3.2}$$

where $\mathbb{1}(\tilde{\mathbf{R}}^{-1}(r_i) \succ \tilde{\mathbf{R}}^{-1}(d_i))$ denotes the indicator function which maps to 1 if $\tilde{\mathbf{R}}^{-1}(r_i) \succ \tilde{\mathbf{R}}^{-1}(d_i)$ is true and to 0 otherwise.

Pairwise preference labels can also be produced automatically based on an already available summarization dataset with a simple strategy. Let set D contain all source sentences (excluding sentences which are contained in the reference summaries) and let R contain all sentences from the reference summaries. The required preference can be generated by sampling pairs of sentences d_i, r_i such that $d_i \in D$ and $r_i \in R$. Based on these pairwise preferences, the systems' ability to distinguish good sentences stemming from a reference summary and bad sentences stemming from a source document can be evaluated.

Since pairwise preference prediction is a special case of ranking generation, it enjoys all previously discussed advantages.

3.6 Summary

In this chapter, we discussed different problems in which estimating information importance is required to perform well in the given problems. The most prominent example is automatic summarization, which is discussed in Section 3.1. Automatic summarization aims at reducing the length of a text document or document collection while retaining the most important information. Based on the definition of importance given in Chapter 2, we were able to provide formal definitions of optimal summaries. We also discovered a close connection between the proposed abstract framework and the Pyramid method, which can be viewed as a concrete implementation of the abstract framework.

Extractive summarization, which has been discussed in Section 3.3.1, is one solution approach to generate summaries automatically. The key idea of extractive summarization is to generate a summary based on sentences that have been extracted from the input documents. Greedy selection is a frequently used algorithm to perform extractive summarization. The most important advantages of greedy selection algorithms are the simplicity of the algorithm and its great computational performance. Low computational complexity is indispensable for a large scale application of summarization, in particular, if large document sets have to be summarized. More costly methods such as integer linear programming are not appealing in these scenarios. We discussed several variants of greedy selection algorithms.

We discussed that automatic summarization has several shortcomings if we want to evaluate the importance estimation capabilities of systems. To mitigate these shortcomings, we presented in Section 3.5 more appropriate alternatives. The example in Chapter 1 can, for example, be used to produce data for alternative methods. The alternative evaluation methods have not been considered by the summarization community so far and may improve the analysis of extractive summarization methods in future works. We demonstrate the applicability of the ranking and the preference prediction tasks in evaluation experiments in Section 6.3.5.

4 Context-free Information Importance Estimation

So far, we have provided formal definitions for information importance (Chapter 2) and have discussed how information importance estimation abilities can be assessed (Chapter 3). The focus of this chapter is a new kind of information importance estimator, which is fundamentally different from prior work. A key contribution of this chapter is the motivation for context-free information importance estimators by answering the question: **Why do we need context-free information importance estimation?**

In Section 4.1, we first define the term *context-free information importance estimation*. The key idea is to estimate the importance of an information nugget without considering the context it appears in. In this thesis, the context of an information nugget always refers to the information nuggets with are in its neighborhood. In text summarization, this is usually a text document.

Context-free information estimation is a distinguishing factor of this thesis, which has not been explored before. The main reason for this observation is that usually only single texts genres are considered by prior work. In particular, newswire documents have been used for summarization frequently where features such as information location and information frequency can be used to estimate information importance. Since prior approaches depend on these document-derived features, they fail in situations in which document-derived features are not available or misleading. One example of such a situation is the U.S. presidential election discussed at the beginning of the introduction of this thesis (see Chapter 1). in

In Section 4.1.1, we first discuss why newswire summarization works the way it does. In Section 4.1.2, we explain why it does not work for another domain. We consider a transcript of a soccer match as an example in which frequently used features such as sentence position, and information frequency cannot be used to estimate the importance of information reliably. The key idea of this thesis is to foster the development of methods that have prior common sense knowledge which can be used to summarize the soccer transcript, for example. In this case, the knowledge that scoring goals is important in soccer is essential to generate a good summary of the soccer transcript. Furthermore, models have to know that the information which team won is also important.

In Section 4.2, we discuss various importance signals used by summarization systems and conclude that relying on these features does not allow a reliable estimation of information importance since none of these features is causally linked to information importance. Hence, the models fail in scenarios where these features cannot be used to estimate information importance. The models are, for example, not able to solve the U.S. presidential election task provided in Chapter 1 since no document-derived features can be used to estimate the importance of the three provided information nuggets. This example is by no means only a theoretical problem. The majority of important information on the Internet might be contained in noisy, unstructured non-newswire articles. In Section 4.2.1, we discuss the information location as one very frequently used contextual importance signal. Section 4.2.2 discusses methods that are based on computing the frequency of information in a document based on sentence similarity measures. In Section 4.2.3, we review topic and cluster-based summarization works. Section 4.2.4 discusses methods based on *n*-gram frequencies. More recent methods, which learn representations of documents are also contextual importance estimators. They are briefly discussed in Section 4.2.5. Strictly speaking, abstractive summarization systems are not information importance estimators. Hence, we only discuss abstractive systems briefly in Section 4.2.6.

Section 4.3 concludes this chapter. The lack of context-free information importance estimation as defined in this chapter motivates the exploration of this new kind of systems in the next part of this thesis.

4.1 Contextual and Context-free Information Importance Estimation

We start by precisely defining what we mean by 'context-free information importance estimation'. As the name already suggests, it is a special kind of information importance estimation. In Chapter 2, we have discussed that information importance can only be defined in a meaningful way while a concrete receiver is considered. Furthermore, we have discussed that the importance of information depends on the receiver and may change if another receiver is considered. Hence, importance estimation is inseparably connected to its receiver. Consequently, we do not refer to the receiver of the information when we talk about the context of the information.

The context of an information nugget is in this thesis defined as all the information which appears in the neighborhood of the information nugget at hand. In text summarization, this is usually a text document or a collection of text documents.

Definition 20 (Semantic Context). Let $\mathfrak{c}$ be the semantic content of a text as defined in Definition 12. Let information nugget $\mathfrak{i}$ be contained in a medium $\mathfrak{m}$ or a set of media $\{\mathfrak{m}_1, \dots, \mathfrak{m}_n\}$ (i.e., $\mathfrak{i} \in \mathfrak{c}(\mathfrak{m})$ or $\mathfrak{i} \in \bigcup_{j=1}^{n} \mathfrak{c}(\mathfrak{m}_j)$). We call $\mathfrak{c}(\mathfrak{m}) \setminus \mathfrak{i}$ or $\bigcup_{j=1}^{n} \mathfrak{c}(\mathfrak{m}_j) \setminus \mathfrak{i}$ **semantic context** of $\mathfrak{i}$.

Similarly, we can define the syntactic context in a text document or a collection of text documents as follows.

Definition 21 (Syntactic Context). Let $\mathfrak{p}$ be a partition function as used in Definition 18. Let a sentence (or text snippet) $\mathbf{s}$ be contained in a medium $\mathfrak{m}$ or a set of media $\{\mathfrak{m}_1, \dots, \mathfrak{m}_n\}$ (i.e., $\mathbf{s} \in \mathfrak{p}(\mathfrak{m})$ or $\mathbf{s} \in \bigcup_{j=1}^{n} \mathfrak{p}(\mathfrak{m}_j)$). We call $\mathfrak{p}(\mathfrak{m}) \setminus \mathbf{s}$ or $\bigcup_{j=1}^{n} \mathfrak{p}(\mathfrak{m}_j) \setminus \mathbf{s}$ **semantic context** of $\mathbf{s}$.

As discussed above, the person how receives the information is not considered as context according to this thesis. Since we do not consider the receiver as context, all from the receiver derived features are consequently also not considered to be context. In particular, the prior knowledge of a receiver is implicitly specified as soon as the receiver of information is considered. Furthermore, the point in time in which a piece of information is received or the location where a piece of information is received is also not considered as context. In the following, we do not differentiate between semantic and syntactic due to simplicity.

Definition 22 and Definition 23 specify contextual and context-free information importance estimators, respectively. Without loss of generality, both definitions only consider one input document X, but could be easily extended to multiple input documents.

Definition 22 (Contextual Information Importance Estimator). Let information nugget $\mathfrak{i}$ be contained in medium $\mathfrak{m}$. Function $\dot{u}$ is called **contextual information importance estimator** if $\dot{u}$ estimates the information importance of $\mathfrak{i}$ based on its context.

Definition 23 (Context-free Information Importance Estimator). Let information nugget $\mathfrak{i}$ be contained in medium $\mathfrak{m}$. Function $\dot{u}$ is called **context-free information importance estimator** if $\dot{u}$ estimates the information importance of $\mathfrak{i}$ without considering its context.

Intuitively, a contextual information importance estimator can be written as function which is parameterized with the context of the information (i.e., $\dot{u}_{\mathfrak{m}}(\mathfrak{i})$) whereas a context-free information importance estimator is not parameterized with the context (i.e., $\dot{u}(\mathfrak{i})$). Typical contextual information importance estimators compute the position of sentences or the frequency with which information appears in a document collection with the help of the context. Context-free information importance estimators do not use such importance signals which depend on the context.

Aiming at developing context-free information importance estimation systems is appealing for several reasons.

First, humans are able to perform context-free information importance estimation because they have a good understanding of information and its impact. Following the example in the introduction, humans know that it is essential to many people who won the U.S. presidential election in 2016 and that many people will be interested in obtaining this information. Humans are also able to estimate that the certification date is not crucially important to many people.

Second, context-free information importance estimation removes assumptions that are made by contextual information importance estimators. Contextual information importance estimators assume that the importance of information can be estimated by analyzing the surrounding context. Some summarization scenarios might meet this assumption to some extent. In general, however, it might not be sufficient to analyze the context to estimate the importance of information. In the example in the introduction, the context cannot be used to estimate the importance of information nuggets. Domain knowledge about the U.S. election is essential to estimate importance in this scenario.

Third, the definition of information importance in Definition 23 also specifies what is required to estimate the importance of information. According to the definition, only a target audience and an information nugget are necessary to estimate its importance. All other signals might be correlated to estimate information importance. They are, however, not causally linked to information importance. Whenever correlated heuristics but not causal features are used for prediction, there will be situations in which the prediction is incorrect.

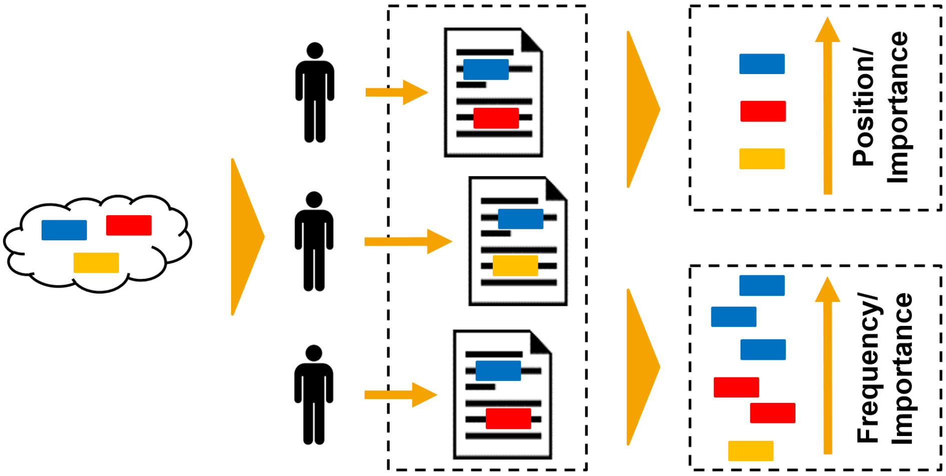


Figure 4.1.: Illustration of journalists who tend to report more important information more frequently and tend to write important information at the beginning of articles.

We analyze in the following section why contextual importance estimators have been so successful in summarization and why they may not be sufficient to summarize heterogeneous and noisy documents.

4.1.1 Why Newswire Summarization Works

We proposed in the last section to develop context-free information importance estimators. However, most summarizers developed so far are contextual summarizers which heavily rely on importance signals derived from the context (see Section 4.2 for more details). Why are contextual summarizers so prominently developed?

To understand why newswire summarization works, we have to take a step back and think about the generation process which generates newswire articles. To this end, we illustrate three journalists writing news articles in Figure 4.1.

All three journalists discovered three new information pieces. For example, we can consider the three information pieces from the introduction. Their job is to write news articles with which they report the new information nuggets to the public. The journalists can choose from three different information nuggets blue, red, and yellow. Journalists do, however, not report all the information gained to the public but filter the information before. We illustrate this in Figure 4.1 by showing written news articles which only contain two information nuggets in the middle. Journalists understand the world they live in and are able to estimate the importance of information. They know which information nuggets are interesting to their readers. Every journalist can be considered to be a context-free importance estimator. Every journalist might have some biases, and their importance estimation also might not always be perfect. Hence, we can consider them as a biased and noisy ensemble of context-free importance estimators.

The first implication of this scenario is that more important information is contained more frequently in the set of generated news articles since the journalists include more important information with a higher probability. In the illustration, the blue information is considered most important by the ensemble of the journalists. The red information is only considered important by two journalists. The yellow information ranks last. Due to this implication, it is a matter of fact that the importance of information correlates with the frequency of information nuggets in document collections. Hence, the journalists introduce a feature into document collections which can be exploited by summarization systems. The positive correlation has already be observed earlier (Nenkova & Vanderwende, 2005; Zopf, Loza Mencía, & Fürnkranz, 2016a) and works in particularly well in multi-document summarization.

A second implication of journalists writing news articles is that important information tends to appear at the beginning of news articles. This fact follows from the fact that journalists follow a particular writing pattern when they write news articles. The pattern is called inverted pyramid[1] and suggests to report the most important information at the beginning of an article and less important information at the end. Again, this explains why sentence position is a very strong importance signal that can be exploited by summarization systems (Zopf, Loza Mencía, & Fürnkranz, 2016a). This fact is most important in single-document summarization in which extracting the first n sentences is sometimes an almost unbeatable baseline.

Hence, summarization systems that exploit these two facts can produce summaries without actually estimating the importance of information by themselves and can instead rely on the importance estimates provided by journalists. In the U.S. presidential election example in the introduction, no pre-filtering has been performed by journalists. The task in this example was basically to replace the journalist instead of analyzing and summarizing the output of journalists. The information nuggets appeared only once and did not have any order. Hence, it was not possible to use these two features to estimate the importance of the provided information. Instead, it was required to understand the information truly and to estimate its impact on the readers.

4.1.2 Transcript of a Soccer Match

In addition to the example in the introduction, we consider a transcript of a soccer match as an additional counter-example in the following.

Figure 4.2 illustrates the transcript of a soccer match between two teams x and y. The commentator reports events in the game such as player a passes the ball to player b substitution of players, free kicks, or yellow cards. All these events are, however, rather unimportant in a soccer match. Most important are goals and the result of the match. In the transcript illustrated in Figure 4.2, one goal is scored approximately in the middle of the game. After the game is finished, the commentator reports that team x won the match. This information is usually the most important information since it implies, for example, that the team advances to the next round in a knockout tournament. Additional information not included in the illustration are details about players, the location of the match, the number of spectators, and the names of the team captains.

[1] https://en.wikipedia.org/wiki/Inverted_pyramid_(journalism)

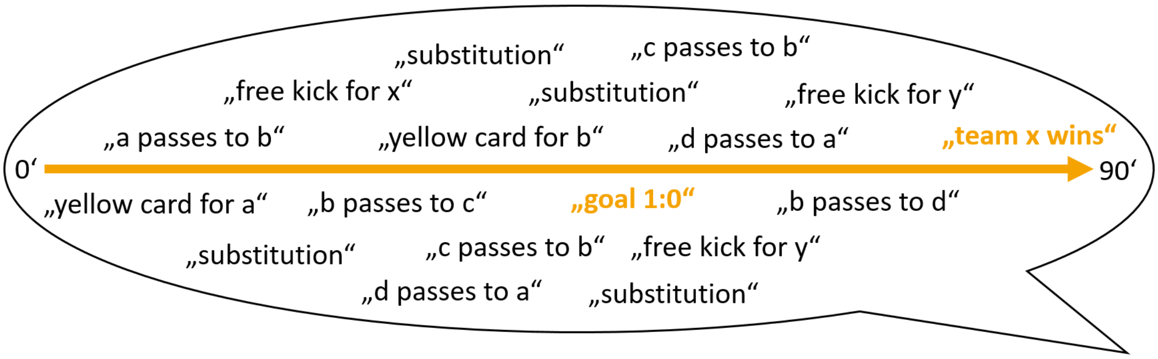


Figure 4.2.: Illustration of a soccer transcript. In soccer, events such as substitutions, free kicks, goals, and the end of the match are reported by commentators. The most important information, namely the scored goal and the result of the match are highlighted.

In the transcript, importance information does not appear frequently. On the contrary, most important events such as goals are rather rare, and frequent events such as passing the ball occur very frequently. The same signals that have been used successfully to summarize newswire articles are not helpful for this document. Newswire articles, however, do only contain a small fraction of important information. Hence, it is important to develop summarization systems which are able to extract the most important information also from more heterogeneous sources. For the soccer transcript, the idea of context-free information importance estimation suggests that the summarizing system should have domain knowledge about soccer. Hence, it should know that goals are important and that reporting the result of a soccer match is important.[2]

4.2 Prior Work on Contextual Information Importance Estimation

In the following, we review several importance signals which have frequently been used by summarization systems to estimate information importance. For each signal, we briefly explain the signal, the intuition behind it, in which works it has been used, and discuss why this signal cannot reliably be used to estimate information importance. A clear classification of systems according to particular features is not always possible since many works rely on a combination of different features. Therefore, we mention some works in multiple categories and some works only to the category where they seem to fit best.

4.2.1 Information Location

As already mentioned, a very simple contextual feature is the location of information. Let $\text{pos}_X(\mathbf{x_i})$ be the position or relative position of sentence $\mathbf{x_i} \in X$. To compute the position of a sentence within a document, information about the document is required, which means that $\text{pos}_X(\mathbf{x_i})$ depends on X and is therefore a contextual feature according to Definition 22. Hence, utility functions $\dot{u}$ which use the sentence position to estimate the importance of $\mathbf{x_i}$ also depend on X.

[2] We present a first context-free summarizer in Chapter 6.

Information location is a strong feature in newswire summarization where important information is usually located at the beginning of articles. This observation leads to a very simple summarization model called *lead summarizer* (Wasson, 1998). A lead summarizer extracts the first *n* sentences from a source document in single-document summarization and extracts the first sentence of every document in multi-document summarization such that the desired summary length is reached. Brandow, Mitze, and Rau (1995) found that using the lead section of articles leads to good summaries. Wasson (1998) investigates how well leads of news articles are accepted by humans as good summaries. For general news, the analysis shows that 94.1% of the lead sections of general news articles are good summaries. For 120-199 word articles, the lead section is acceptable in 98.2%. This result shows that newswire documents have a common structure that can be easily exploited to generate good summaries. Meade (1997) uses position and length information as features to calculate sentence utilities and finds that the position of a sentence is most valuable. Sentence location is not only used by lead summarizers but also by many other summarization systems as a feature (Christensen et al., 2013; Kedzie, McKeown, & Diaz, 2015; Peyrard & Eckle-Kohler, 2017).

The assumption made when information location is used is that the position of a sentence helps to estimate information importance. This assumption is reasonable if the input documents belong to the newswire genre. Journalists tend to write important information at the beginning of a document. Hence, many summarization systems use sentence position as an important feature to estimate information importance. Similarly, the result of a soccer match is usually reported at the end. Information location can also be used in the transcript to find important information. However, the general assumption that important information is located at the beginning of a document is not correct in soccer transcripts. Hence, using the information location feature requires an adaption to the text genre at hand. Furthermore, goals can be scored at any time. Information location can therefore not be reliably used to find all important information nuggets even if the feature is adapted to the ”soccer transcript” text genre.

The motivational example in Chapter 1 can also be considered to show that information location does not provide a reliable signal for information importance. In the example, we provided three different sentences for which the importance had to be estimated. Since the sentences do not have a fixed order but can be ordered arbitrarily, sentence position cannot be used in this situation to estimate information importance.

Moreover, a system can derive the position of a sentence and use it to estimate information importance without reading the sentence. Hence, the sentence position is independent of sentence content. Furthermore, moving a sentence to a different location does not change the information contained in the sentence and cannot be reliably linked to its importance.

We conclude that information location is in general independent from information importance (i.e., there is no causal connection) and can only be used in text genres in which it is known that information location is correlated with information importance. The assumption that information location indicates information importance is in general, not true for input documents from heterogeneous text genres.

4.2.2 Sentence Similarity-based Importance Estimation

A second very frequently used method to estimate information importance is based on sentence similarity. The key idea is to estimate a similarity between sentence pairs in a document or document collection and to select sentences for the summaries based on the estimated similarities. Formally, let X be a set of sentences extracted from input documents and let $x_i \in X$ be a sentence in X for which their importance has to be estimated. A sentence similarity-based importance estimator estimates the importance of x_i by estimating a similarity to other sentences $x_j \in X$. Since the importance of sentences is based on the surrounding sentences, similarity-based methods belong to the class of contextual importance estimators.

A wide range of methods has been proposed so far, which are reviewed in detail in the following. Note, however, that most of the reviewed papers aim at generating a good summary which is not necessarily the same as extracting the most important information from a document.

Similarity-based Methods

Methods based on sentence similarity assume that the importance of sentences can be estimated by investigating pairwise similarities between sentences. A well-known work is the maximum marginal relevance approach by Carbonell and Goldstein (1998) (see Algorithm 6 for a pseudo-code description). The initial idea of MMR was that a sentence is more important if it has a high similarity with a given query indicating the information need of a user. The idea has been extended to estimate the similarity with sentences in input documents instead of estimating the similarity with a query.

Kågebäck, Mogren, Tahmasebi, and Dubhashi (2014) produce vector representations of sentences by adding word vector representation and an unfolding recursive auto-encoder (Socher, Huang, Pennington, Ng, & Manning, 2011). Word2Vec (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013) and vector embeddings (Collobert & Weston, 2008) have been used in the experiments. Kågebäck et al. (2014) test two different sentence similarity measures. First, the cosine similarity of the produced sentence representations is computed. A second similarity measure computes the complement of the Euclidean distance. The complement is used to transform the distance into a similarity. Similarly, Yin and Pei (2015) use a convolutional neural network to learn sentence representations, which are used to compute the similarity of sentences with a Cosine similarity.

Similarly to sentence position, similarity is a frequently used feature in newswire summarization. In particular, in multi-document summarization it is often used because sentences that occur frequently (i.e., having a high average similarity with other sentences in the input documents) usually correlates with importance to some extent. Sentences with high average similarity are considered to be *central* in a document or document collection. Centrality-based approaches work well for the newswire genre because journalists tend to include more important information more frequently in news articles (Zopf, Loza Mencía, & Fürnkranz, 2016a). Following the example in Chapter 1, many journalists are likely to report that Donal Trump won the election and only a few journalists report the fact that the Congress certifies the results of the election on January 6, 2017. This observation leads to sets of news documents in which frequency indeed correlates with importance. The positive correlation of frequency and importance results from the fact that journalists are already estimating the importance of information when they write news articles. It can be argued that methods that rely on information centrality do not estimate importance but only pick up the estimates introduced by journalists. If input documents do not belong to the newswire genre, it might not be true that the most important information is also the most frequent information. The introductory example provides a counter-example for such a situation. All three information nuggets available in the input occur only once. A system that relies on the assumption that centrality can be used to detect importance fails in this situation. The previously mentioned soccer transcript provides an even more clear counter-example. Important events in soccer matches such as goals and penalties are rare, whereas unimportant events such as passing the ball or minor heavy fouls occur very frequently. Importance correlates negatively with centrality in this text genre. Hence, methods which extract central information may not be able to perform well. Similarly, centrality-based methods fail in all text genres in which the assumption that centrality indicates importance does not hold. This might be true for many, more heterogeneous input document sets.

We conclude that centrality is a good indicator of importance in the newswire genre since journalists tend to include more important information more likely. In other genres, this assumption might not hold, and domain knowledge is required to estimate the importance of information reliably.

Graph-based Methods

Graph-based methods can be viewed as an extension of similarity-based methods. The key difference is that not only the similarity to other sentences is measured to estimate the importance of a sentence, but additional properties of other sentences are also considered. To this end, graphs are constructed based on sentence similarities. Directed graphs can be constructed if the ordering of sentences in the documents is also considered. Frequently used approaches are PageRank (Brin & Page, 2012) and HITS (Kleinberg, 1999). TextRank (Mihalcea, 2005; Mihalcea & Tarau, 2004), for example, is a saliency-based summarization system that models sentences as nodes in a graph, where the strength of the connections between the nodes is determined by the similarity of the sentences measured by means of syntactic word overlap. TextRank uses the PageRank algorithm to find sentences that are well connected in a graph that has been constructed based on sentence similarities. Mihalcea (2004) extends this work by investigating the effect of different graph ranking algorithms. Erkan and Radev (2004) propose a graph-based method which uses intra-sentences cosine similarity to compute an adjacency matrix to represent the sentences as a graph. The most central sentence is considered to be the most important sentence. Parveen and Strube (2015) use a bipartite graph to represent a document. The so-called topic graph has two sets of nodes, one containing sentences and one containing topics. To rank the sentences, they apply the HITS algorithm. Wan, Yang, and Xiao (2007) assume that sentences and words are important if they are heavily linked with other words and sentences. Furthermore, they assume that sentences are important if they contain important words and words are important if they are contained in important sentences. They connect both assumptions to PageRank and HITS, respectively.

4.2.3 Topic / Cluster-based Summarization

The key idea of topic-based and cluster-based summarization models is to find subtopics or clusters in the document collection at hand. Based on created clusters, sentences can be extracted according to different strategies for mode-focused, diversity-focused, and centrality-focused summarization.

Summarist (Hovy & Lin, 1999) specifies topic identification as one key element for summarization. Topic signatures are words that can be used to identify the presence of a particular (sub-)topic. Topic importance is in turn estimated by topic centrality. Topics that are most central in the documents are considered to be most important. C.-Y. Lin and Hovy (2000) automatically acquire topic signatures from data. Celikyilmaz and Hakkani-Tur (2011) and D. Wang, Zhu, and Li (2009) use topic models for summarization.

M. Liu, Li, Wu, and Lu (2007) use DBSCAN to cluster event terms which describe events in the source documents. They investigate two summarization strategies. The first strategy picks one event term from every cluster, and the second strategy picks all terms from the biggest cluster. Sentences are selected to cover as many selected terms possible. Both strategies are similar to the diverse and the representative summarization strategy, respectively, with the difference that diversity and representativity are not defied with sentence similarity but with event term similarity.

Nomoto and Matsumoto (2001a) investigate *diversity-based* summarization. The key idea of this work is to extract sentences that cover many different subtopics in a document. To this end, a clustering algorithm is applied to a document to identify different subtopics. From every subtopic, the most representative sentence is extracted and added to the summary. The size of the cluster (i.e., the saliency of the corresponding topic) is not considered when sentences are selected. Hence, the model creates summaries that cover many different areas of source documents. Another version of diversity-based summarization has been introduced by Attokurov and Bayazit (2014). They build a tree with a hierarchical agglomerative clustering algorithm and prune the resulting tree the optimal tree pruning algorithm BFOS (Breiman, Friedman, Stone, & Olshen, 1984; Chou, Lookabaugh, & Gray, 1989). The sentences in the leaf nodes are subsequently included in the summary.

Nomoto and Matsumoto (2001a) evaluate summarization models by evaluating how well information retrieval tasks such as document retrieval or document classification can be performed by using the created summary as a surrogate for the original documents. The work uses K-means clustering to find different subtopics in the documents and applies a diversity-based selection based on the created clusters. Nomoto and Matsumoto (2001b) use different probabilistic decision trees to classify sentences in the categories important and unimportant. A probabilistic decision tree does not assign each instance to a single class but distributes each instance among different classes. The models use mainly length- and location-based features. The probabilistic decision trees are applied to the original documents as well as to clusters found by Nomoto and Matsumoto (2001a).

Summa (Christensen, Soderland, & Bansal, 2014) generates a hierarchy of summaries that can be explored by a user. A child summary adds more detailed information to the general information contained in its parent. To generate the hierarchy, a clustering algorithm first clusters the sentences and summarizes the clusters in a second step. For the summarization, a linear regression model is trained, which finds sentences more salient that contain more frequent nouns and verbs based on the findings by Christensen et al. (2013).

A. J. Tixier, Malliaros, and Vazirgiannis (2016) find that keywords can be found among influential nodes rather than eigenvector- or random walk-related centrality measures as, for example, used by PageRank. The findings are presented in an interactive web application (A. Tixier, Skianis, & Vazirgiannis, 2016).

Shang et al. (2018) also use a graph-based model of the text to identify important words. Additionally, they cluster and compress sentences that are similar before a budgeted submodular maximization approach selects sentences.

W. Li, Wu, Lu, Xu, and Yuan (2006) build a graph containing event terms and named-entities. Different perspectives of similarities are considered. Named-entities are considered to be similar if they appear in the same events frequently and vice-versa. Another similarity between events is based on WordNet. PageRank is applied to estimate the importance of the nodes, meaning that central nodes are considered to be more important than non-central nodes.

Gao, Li, and Zhang (2013) summarize Twitter trending topics sequentially by publishing one sub-summary to every subtopic or sub-aspect of the trending topic. Subtopics are detected by observing surges in the Twitter stream. For every subtopic, a representative tweet is selected as summary.

Ye, Chua, and Lu (2009) represent Wikipedia pages in terms of contained concepts (i.e., terms for which Wikipedia articles exist). Sentences are considered to be more important if they contain concepts with higher CF-IDF (concept frequency inverse document frequency).

Ng, Chen, Kan, and Li (2014) construct timelines based on the source documents and extends the SWING summarizer (Ng, Bysani, Lin, Kan, & Tan, 2012). The idea is that a timespan should be considered to be more important if more events describe it. Timespans that contain more events are therefore considered to be more important, and sentences describing more important timespans are considered to be more important as well.

4.2.4 *n*-gram Frequency-based Importance Estimation

Methods based on *n*-gram frequencies use the frequencies of *n*-grams in the documents to estimate information importance.

Early works in summarization (Edmundson, 1969; Kupiec, Pedersen, & Chen, 1995; Meade, 1997) used TF-IDF scores to identify important information. (Nenkova & Vanderwende, 2005) analyze the impact of word frequencies and finds that frequency in the input documents is strongly indicative of whether a word will also appear in a reference summary. They use this observation to create the SumBasic summarizer. SumBasic computes a weight for each word in the input document based on its frequency. Sentences are selected iteratively according to the average word probability of the words which are contained in the sentence. Yih, Goodman, Vanderwende, and Suzuki (2007) extend SumBasic by also incorporating position information and using a machine learning model to combine the different importance signals. The ICSI summarization system (Gillick et al., 2008; Gillick et al., 2009) is a well-known representative of *n*-gram distribution based summarization which uses, similar to the work presented by Nenkova and Vanderwende (2005), the frequency of bigrams to estimate information importance. Schluter and Søgaard (2015) extend ICSI (Gillick et al., 2008; Gillick et al., 2009) by testing other text annotations than bigrams and finds that bigrams work only best in newswire datasets whereas other annotations such as frames work better in other text genres. NeATS (C.-Y. Lin & Hovy, 2002a) scores sentences based on the saliency of contained key concepts. Key concepts are unigrams, bigrams, and trigrams that frequently appear in the source documents. Furthermore, three filters based on sentence position, stigma words, and MMR-based scoring are used to filter sentences that do not meet the criteria defined by these filters. H. Lin and Bilmes (2011) use cosine similarity based on TF-IDF scores of bigrams. Sentences that contain more frequent bigrams are considered to be more important.

C. Li, Qian, and Liu (2013) use a regression model to learn to estimate the frequency of bigrams in the reference summary instead of calculating the weight of bigrams based on input document frequencies such as (Gillick et al., 2009). (C. Li et al., 2013) use word- and sentence-level features such as bigram frequency, bigram-topic similarity, and position and length of sentences that contain the bigram. Given the frequencies, an ILP selection process selected the sentences such that the most frequent bigrams are included in the automatically generated summary.

Louis (2014) defines new information nuggets as summary-worthy based on Bayesian surprise (Itti & Baldi, 2005) for generic and update summarization. Sentences are greedily selected according to their Bayesian surprise, which is computed based on word unigrams. The higher the surprise is to see a text with respect to a background corpus, the higher is the score of the sentence. Since the model estimates the surprise based on the input documents, it is also a contextual importance estimator. The model could, however, easily be converted into a context-free information importance estimator by estimating the surprise based on individual sentences. How well the model would perform is unknown.

Ouyang, Li, and Lu (2009) build a hierarchy of words that appear in source documents. The parent of a word is the word that appears more frequently than the child node and has the highest pointwise mutual information with respect to all other words in the source document. A word is considered to be important if the child's words are important. Initial importance is defined to be the log-frequency of the words. Scores of sentences are computed by averaging the word importances given by the computed tree.

Gupta, Nenkova, and Jurafsky (2007) examine two different ways to determine word importance, both of which are based on the frequency of the word, pure frequency, and log-likelihood ratio (Manning & Schütze, 1999). They find that the log-likelihood ratio works better not because it defines a more accurate assignment of importance than word frequency but because it makes it possible to distinguish between non-descriptive and descriptive words, so-called signature terms, (C.-Y. Lin & Hovy, 2000).

G-Flow (Christensen et al., 2013) is a joint model for selection and ordering that balances coherence and salience of sentences. G-Flow trains a linear regression model based on surface-level features such as sentence position, number of people mentions, sentence length, whether the sentence contains an amount (money), and number of other sentences that mention a noun or a verb in the given sentence. The last two features are found to be most indicative for sentence importance, meaning that a high overlap of verbs and nouns with other sentences indicates important sentences.

PriorSum (Cao, Wei, Li, et al., 2015) models importance estimation as regression task and uses features learned by a convolutional neural network in addition to term frequency features and sentence position to estimate sentence utilities.

Like approaches based on sentence similarity, n-gram frequency-based methods have demonstrated good performance in newswire summarization. However, they are not suited to estimate information importance in the U.S. presidential election example or for the provided soccer transcript because the assumption that frequency correlates with importance is not satisfied.

4.2.5 Learned Text Representations

More recent approaches learn a representation of the input document. Cheng and Lapata (2016), for example, use a convolutional neural network to encode sentences. On top of the sentence representations, they use a recurrent neural network to model sentence dependency. Hence, this summarization system models contextual information explicitly. SummaRuNNer (Nallapati, Zhai, & Zhou, 2017) uses a two-layer bidirectional recurrent neural network for sentence and document representation. The first layer uses word embeddings to produce sentence representations, and the second layer uses sentence representations to produce document representations. The document representation is used to predict for each sentence whether or not it should be included in the summary. Since the model uses information not only from the sentence for which a prediction has to be made but information from all sentences in a document, it is a contextual importance estimator.

4.2.6 Abstractive Summarization Systems

In recent years, interest in abstractive summarization has been increasing. Compared to extractive summarization, abstractive systems do not extract text parts such as phrases or sentences from the input documents but produce summaries from scratch. Since abstractive models do not extract sentences from the input documents, they are not extractive summarizers and not information importance estimators according to Definition 22. Hence, they are out of the scope of this section and will only be reviewed briefly.

Sjöbergh (2007) presents a simple abstractive summarization system that selects words greedily based on a Markov model of the input document. The produced summaries are merely concatenations of n-grams and cannot be considered to be good summaries. However, the model achieves super-human ROUGE recall scores, which indicates that ROUGE is not a reliable evaluation method for abstractive summarization models.

Rush, Chopra, and Weston (2015) use an encoder-decoder system to learn to predict the headline of a newswire article based on the first sentence of the article. Chopra, Auli, and Rush (2016) replaces the language model decoder used by Rush et al. (2015) with a recurrent neural network. Nallapati, Zhou, dos Santos, Gulcehre, and Xiang (2016) use additional annotation such as part-of-speech tags, named-entity annotations, and TF-IDF scores in addition to word embeddings as features to encode the text. To reduce repetitions in the output, See, Liu, and Manning (2017) use a covering mechanism to attend to different parts of the input. SWAP-NET (Jadhav, 2018) models the importance of words and sentences by training pointer networks that learn to point on words and sentences that are likely to appear in the summary. This model can be considered to be a supervised version of the model proposed by Zopf, Loza Mencía, and Fürnkranz (2016a). Similarly, Hsu et al. (2018) combine an extractor and an abstractor network. The former is trained as a binary classifier to predict whether or not a sentence is contained in the summary (hard attention). The probabilities predicted by the extractor can also be used as soft attention.

Arumae and Liu (2018) train a reinforcement learning model to include answers to questions automatically generated based on reference summaries. A set of sentences is sampled, and feedback is received. The model receives a high reward if many keywords answering the generated questions are contained in the generated summary. A representation of the source document is learned based on a Bi-LSTM initialized with GloVe word embeddings (Pennington, Socher, & Manning, 2014).

It has also been demonstrated that abstractive summarization and extractive summarization are not mutually exclusive (P. J. Liu et al., 2018). Encoder-decoder models produce summaries contextual and not context-free since they process the input text completely before they produce output.

4.3 Summary

In this chapter, we first defined the term 'context' and defined what contextual and context-free information importance estimators are. Context-free information importance estimates the importance of information without considering the surrounding text. Context-free information importance estimation is required to reliably estimate information importance in challenging scenarios such as the example given in the introduction (see Chapter 1). Systems that rely on the context of an information nugget to estimate its importance are not able to solve this problem since the context is arbitrary or no context is provided at all. This fact motivates research for context-free information estimation and thereby the development of context-free information importance estimators in Part II of this thesis. Subsequently, we reviewed summarization systems and organized them according to the features they use to estimate information importance. The key conclusion of this review is that most summarization systems proposed until today are contextual information importance estimators.

We conclude that many summarization models use genre-specific signals to detect information importance. Most of the signals are either location- or frequency-based. This observation results from the fact that most summarization systems are built to work well in the newswire genre. The use of signals based on information position and information frequency works well in newswire summarization since journalists tend to write the most important information at the beginning of an article and include the most important information in many articles. Hence, journalists estimate information importance and write articles such that summarization systems can pick up these signals to retrieve the information importance estimates made by journalists without understanding the content. A good example of such a signal is sentence position, which can be computed without analyzing the information at all. Similarly, signals based on frequency can be computed without analyzing the information itself.

We have provided two counter-examples in this thesis in which relying on information frequency, or information position is not sufficient to detect information. In the U.S. presidential election example, all three information nuggets appear only once, and the order of the information nuggets is arbitrary. Hence, frequency and position cannot be used to detect important information. In the soccer transcript, important information can occur at arbitrary positions and are rather rare than frequent. Many more situations and text genres can be found in which information frequency and position are no reliable indicators of importance. Other interesting genres are educational texts (Benikova, Mieskes, Meyer, & Gurevych, 2016), chat protocols (Zhou, Hovy, & Rey, 2005), forum threads (Verberne, Krahmer, Hendrickx, Wubben, & van Den Bosch, 2018), web pages (Falke & Gurevych, 2017) and even mixtures of genres (Zopf, Peyrard, & Eckle-Kohler, 2016). To make use of the information in these unstructured information sources, we need information importance estimators that are able to estimate the importance of information based on the information itself, i.e., context-free, and not based on genre-specific contextual importance signals.

Part II

Machine Learning for Context-free Information Importance Estimation

After motivating context-free information importance estimation in Part I of this thesis, we focus on the question how machine learning models can be built that make use of the concept of context-free information importance estimation. We need machine learning to solve this problem because it turns out to be a very difficult task to explicitly program machines to estimate information importance. Machine learning aims at avoiding the need for writing task-solving algorithms by developing algorithms that can learn to perform tasks based on experience. It requires three essential ingredients: data, learning algorithms, and evaluation methods.

A common way to provide experience in machine learning are datasets, which contain samples of the task at hand. In Chapter 5, we discuss how summarization datasets for information importance estimation can be constructed. In particular, we focus on cost-efficient construction of heterogeneous datasets that require context-free information importance estimation capabilities since input documents may not provide easy-to-exploit importance signals.

In Chapter 6, we discuss how machines can acquire context-free information importance estimation abilities similar to humans. To this end, we present a fully context-free information importance estimator that learns to estimate information importance by gaining prior domain knowledge. Furthermore, we discuss sentence regression, which is a promising subfield of extractive summarization.

In Chapter 7, we present a new evaluation model which aims at estimating the performance of automatic summarization models. Developing new evaluation models is crucial due to the minor precision of previous models. Contrary to prior models, which make use of reference summaries, our model is trained based on context-free pairwise preferences of sentences. Since generating the training data for the new evaluation method is much cheaper, it is better suited to be used for new and large datasets.

5 Cost-Efficient Creation of Heterogeneous Summarization Datasets

Experience is the first major ingredient to train machine learning models. Consequently, it is crucial to have good datasets to learn from on the quest towards context-free importance estimation. However, datasets are not only required to train models but also to evaluate them. Creating datasets for multi-document summarization is very costly, which prohibits the creation of large training and test dataset. This problem causes a serious limitation for developing new machine learning models. Furthermore, currently available dataset are very homogeneous, which makes summarization rather simple and hides the hard challenges towards robust importance estimation. Consequently, we focus in this chapter on the question: **How can challenging datasets for information importance estimation be created cheaply?**

In Section 5.1, we discuss relevant terminology of this chapter.

In Section 5.2, we review existing corpora for many different kinds of automatic summarization. We start with sentence summarization in Section 5.2.1, which aims at reducing the length of a given sentence while retaining its meaning. Headline generation, which aims at generating headlines of news articles, is discussed in Section 5.2.2. In Section 5.2.3, we discuss one of the most prominent summarization domains, namely single-document summarization. The task here is to generate a short summary of one input document. Contrary, multiple documents have to be summarized in one summary in multi-document summarization. We discuss multi-document summarization in Section 5.2.4. The previously discussed setups usually consider the newswire documents. In Section 5.2.5, other text genres are discussed. Structured summarization aims at generating summaries which are not one written text document but have more structure such as summarization graphs or hierarchical summaries. We discuss this area in in Section 5.2.6. We discuss the limitations of already created corpora in in Section 5.2.7. Most importantly, we find that creating corpora is often costly. Consequently, the resulting corpora are rather small, which is in particular an issue in multi-document summarization. Furthermore, the strong focus on the newswire genre is a very narrow view on the wide range of possible application areas for summarization.

In Section 5.3, we discuss a new corpus construction approach. First, we describe how dataset have been created in the past in Section 5.3.1. Next, we discuss limitations of the traditional approach in Section 5.3.2. We present a new corpus construction approach in Section 5.3.3, and discuss its advantages as well as potential problems in Section 5.3.4.

We use the newly presented approach and demonstrate its applicability in Section 5.4. First, we discuss some additional goals we want to achieve in Section 5.4.1. In Section 5.4.2, we describe how we used the new approach to create *h*MDS. In this implementation of the approach, we use Wikipedia as source for reference summaries and searched manually for corresponding source documents. In Section 5.4.3, we present an analysis of the generated corpus including quantitative results such as its size, lengths of source documents and summaries, as well as its heterogeneity. Section 5.4.4 concludes the presentation of the new corpus construction approach.

In Section 5.5, we improve the cost-efficiency of the presented approach by reducing the required human labor even further. We describe the modifications in Section 5.5.1. Most importantly, we propose to retrieve source documents automatically. In Section 5.5.2, we analyze the newly created corpus called auto-*h*MDS. We performed summarization experiments to provide first reference results for future work. The results are reported in Section 5.5.3.

We summarize this section in Section 5.6.

5.1 Foundations

We define the terms *task* and *dataset*, which are the first essential components of machine learning in this section.

Let $f : \mathscr{X} \rightarrow \mathscr{Y}$ be a (potentially multivalued) function which maps from set $\mathscr{X}$ to set $\mathscr{Y}$. The function f models the task at hand, the domain of f is $\mathscr{X}$ which contains all possible inputs for f and the codomain of f is $\mathscr{Y}$ which contains all corresponding outputs. $\mathscr{X}$ can, for example, contain all real-valued numbers, all grammatically correct sentences in a specific language, or all images containing faces. In these examples, $\mathscr{Y}$ might contain all square-roots of the real-valued numbers, all correct translations of the sentences, and all detectable faces which are contained in the images, respectively. The tasks f in the examples would be ”compute the square-root of the input”, ”translate the input sentence”, and ”tag all faces in the input image”, respectively.

An illustration of a task which is represented by function f can be found in Figure 5.1. In the illustration, we display a continuous function f for simplicity. A task, however, can also be non-continuous or can have non-continuous domains and codomains.

Many interesting tasks, such as computations of square-roots, translation, and face detection contain a vast number of possible input-output pairs. For the three mentioned problem, the size of the datasets might be even infinite if there is an infinite amount of real-valued numbers, input sentences, and images. Even for problems that can be enumerated with a finite number of input-output pairs such as chessboard states and corresponding optimal moves, it can still be infeasible to create datasets that contain all possible pairs.

Hence, datasets that represent a task usually contain only a small subset of all possible input-output pairs. The domain $\mathscr{X}$ of the task reduces to a subset of samples from the task $\mathbf{X} \subset \mathscr{X}$. The codomain reduces in this case as well to $\mathbf{Y} \subset \mathscr{Y}$ since not all outputs can be reached anymore from $\mathbf{X}$. We denote datasets by $\mathbf{D} = (\mathbf{X}, \mathbf{Y})$. Individual input-output pairs are denoted by $(\mathbf{X}_i, \mathbf{Y}_i)$ such that $\mathbf{Y}_i = f(\mathbf{X}_i)$. Figure 5.2 illustrates a dataset which only contains three samples from the task illustrated in Figure 5.1. In this example, the data points are correct samples from the task since they contain the correct output for the given input points. However, this situation does not have to be the case in real datasets. Usually, data

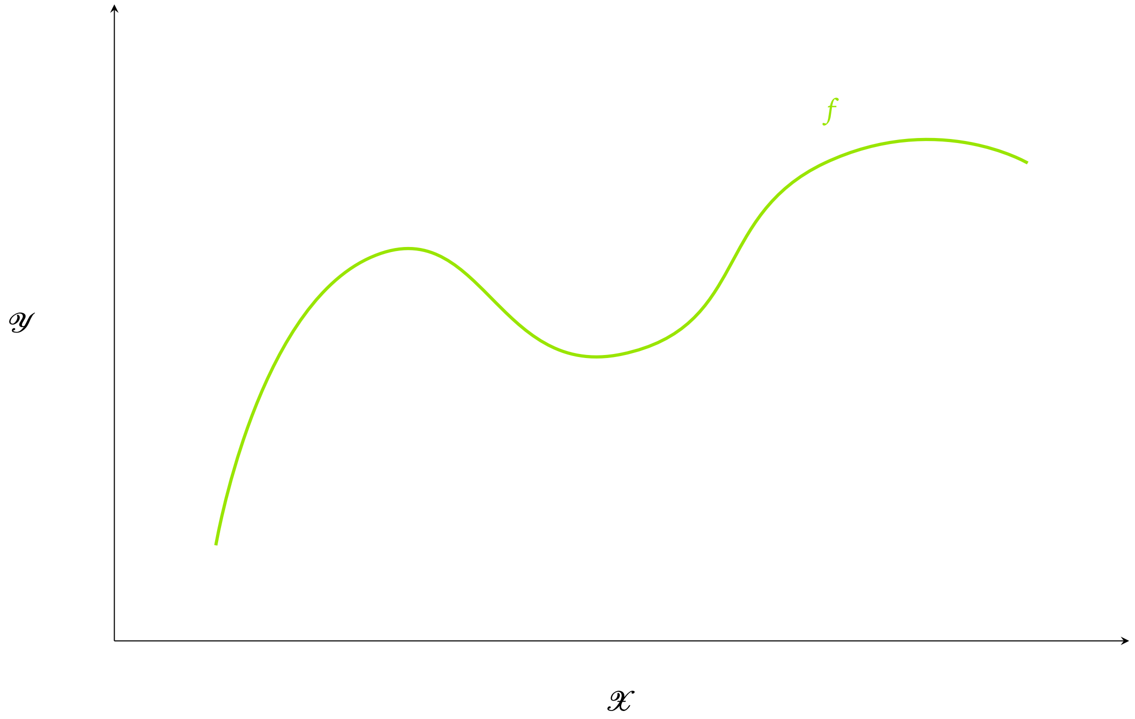


Figure 5.1.: Illustration of a task. The task is illustrated by a function f which maps from input in domain $\mathscr{X}$ to outputs in codomain $\mathscr{Y}$. In this example, both domain and codomain are continuous spaces like $\mathbb{R}$. Furthermore, function $\mathscr{Y}$ is also continuous.

points may contain noise, for example, due to annotation mistakes during the data collection process. The data points are, in this case, not exactly on the illustrated line. Hence, datasets do not contain samples from the true task in practice, but rather samples from a noisy approximation of the task. We omit this fact in the illustration due to simplicity. Furthermore, some features required to describe $\mathbf{X}_i$ correctly might be missing. Consider, for example, a dataset containing properties of people and a classification whether or not they are eligible to obtain a loan. For some people (samples) in the dataset, the property 'age' might be missing due to several reasons. It is not possible to precisely locate the data point in the space of possible data points. We also do not illustrate this practical issue in Figure 5.2.

Datasets are an essential ingredient for machine learning since machine learning algorithms aim at learning tasks (such as f) from samples in datasets $\mathbf{D} = (\mathbf{X}, \mathbf{Y})$.

We review in the following datasets which are already available for task automatic summarization.

5.2 Existing Datasets for Automatic Summarization

Research in automatic summarization has already a long history (Mani, 2001; Nenkova & McKeown, 2011). Hence, many summarization datasets have already been created. Datasets have been used since the very early days of automatic summarization (Edmundson, 1969; Luhn, 1958) for both training and testing supervised and unsupervised machine learning models. We review in the following summarization datasets in detail, starting with corpora for sentence summarization (Section 5.2.1) and

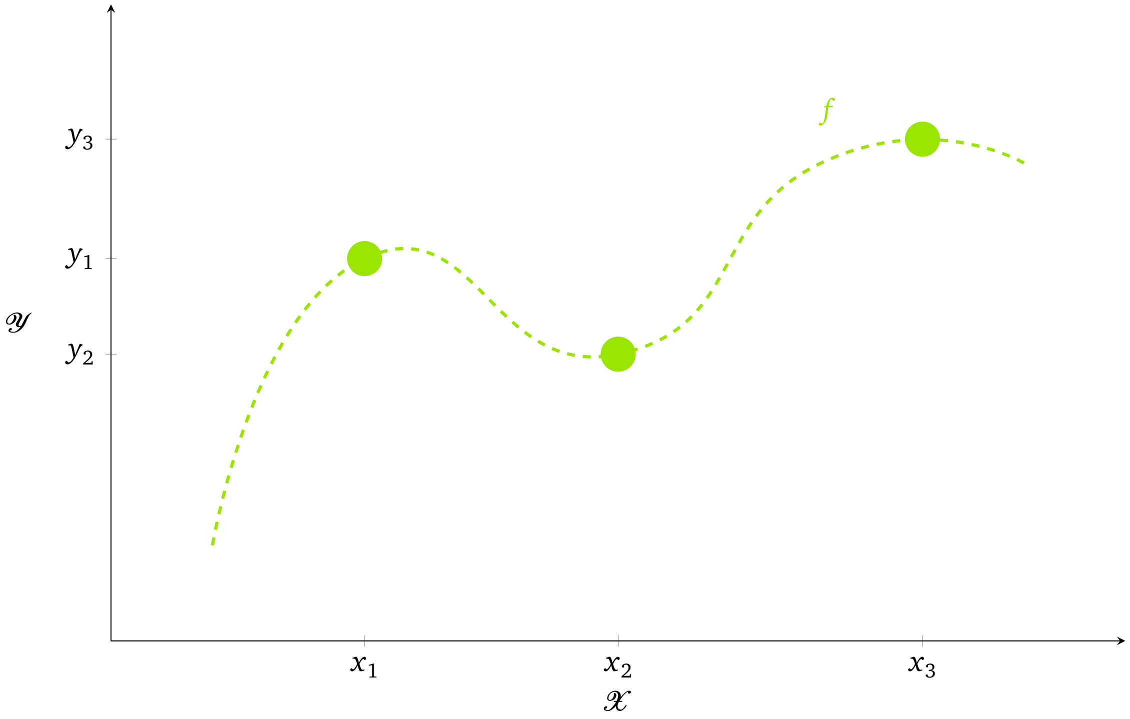


Figure 5.2.: A machine learning dataset for a task specified by f is a finite sample of inputs (e.g., $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3)$) paired with corresponding outputs (e.g., $\mathbf{Y} = (\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3)$) with $\mathbf{Y}_i = f(\mathbf{X}_i)$. Important to note is that datasets usually represent only a small part of the underlying task.

headline generation (Section 5.2.2), followed by single-document summarization (Section 5.2.3) and multi-document summarization (Section 5.2.4) corpora. Table 5.1 and Table 5.2 provide an overview of discussed datasets. A recent overview of summarization corpora has also be created by Dernoncourt, Ghassemi, and Chang (2018).

5.2.1 Sentence Summarization

Sentence compression is a subfield of automatic summarization in which the task is to summarize a single, standalone sentence instead of a text containing many sentences.

The Ziff-Davis corpus[1] is a collection of newswire articles about computer products. A human-written summary accompanies each article. To build a sentence compression corpus, the sentences from the article have been automatically aligned with sentences from the summaries given that some criteria have been met (Jing & McKeown, 1999; Knight & Marcu, 2002). One possible criterion is, for example, that the compressed sentences have to contain words from the sentence to be summarized, in the same order.

The MSR Abstractive Text Compression (MSR ATC) Dataset[2] is a corpus for single sentence and short paragraph (two sentences) compression (Toutanova & Brockett, 2016). Covered text genres are

[1] https://catalog.ldc.upenn.edu/LDC93T3A

[2] https://www.microsoft.com/en-us/research/project/intelligent-editing

Dataset	Type	Genre	Lang.	Src/Topic	Topics	Src Len.	Sum Len.
MSR ATC [1]	generic	multiple	en	1	6,169		
Ziff-Davis [2]	generic	news	en	1	1,067		
[3]	generic	news	en	1	250k		
[4]	generic	news	en	1	575		
PubMed [5]	scientific abstract	scientific	en	1	133k		
arXiv [5]	scientific abstract	scientific	en	1	215k		
NYT [6]	generic	news	en	1	1.8m		
CNN [7]	generic	news	en	1	93k		
Daily Mail [7]	generic	news	en	1	220k		
Newsroom [8]	generic	news	en	1	1.3m	658.6	26.7
LCSTS [9]	generic	news	cn	1	2.4m		

[1] Toutanova and Brockett (2016)
[2] Knight and Marcu (2002)
[3] Filippova and Altun (2013)
[4] Cohn and Lapata (2008)
[5] Cohan et al. (2018)
[6] Sandhaus (2008)
[7] Hermann, Kocisky, and Grefenstette (2015)
[8] Grusky, Naaman, and Artzi (2018)
[9] B. Hu, Chen, and Zhu (2015)

Table 5.1.: Overview of single-document summarization corpora. We provide the common name of each dataset along with a reference as far as possible. Furthermore, we provide information about the type, the text genre, and whether the dataset is multilingual. *Sources/Topic* indicates how many sources per topic are available. *Topics* indicates how many topics the dataset contains. *Source Length* and *Summary Length* indicate the number of words in the source documents and summaries, respectively. If there is a variation in any amount, we provide the average along with the standard deviation in the format "average ± standard deviation".

Dataset	Type	Genre	Lang.	Src/Topic	Topics	Src Len.	Sum Len.
DUC 2001 [1]	generic	news	en	10.2 ± 2.1	60		
DUC 2002 [2]	generic	news	en	about 10	60		
DUC 2003 [3]	focused	news	en	10	30		
DUC 2004 [4]	generic/focused	news	en	10	50+24+50		
DUC 2005 [5]	question focused	news	en	25-50	50		
DUC 2006 [6]	question focused	news	en	25	50		
DUC 2007 [7]	q.focused+update	news	en	25	45		
TAC 2008 [8]	focused+update	news	en	20	48		
TAC 2009 [9]	focused+update	news	en	20	44		
TAC 2010 [10]	guided+update	news	en	20	46		
TAC 2011	guided+update	news	en	20	44		
TAC 2014	citation summary	scientific	en	10	50		
Opinosis [11]	opinion-focused	reviews	en	about 100	51		
Concept Maps [12]	topic provided	heterog.	en	40.5 ± 6.8	30		-
Hier. Sum [13]	generic	heterog.	en	76.5 ± 17.1	10		-
DBS [14]	generic	heterog.	de	9.2 ± 3.2	10		
Live Blog [15]	generic	news	en	70.3	2,655	90.3	48.6
*h*MDS [16]	generic	heterog.	multi	13.9 ± 3.1	91		
auto-*h*MDS [17]	generic	heterog.	multi	8.85	7,316		

[1-4] Over (2001), Over and Liggett (2002), Over and Yen (2003, 2004)
[5-7] Dang (2005, 2006, 2007)
[8-10] Dang and Owczarzak (2008, 2009), Owczarzak and Dang (2010)
[11] Ganesan, Zhai, and Han (2010)
[12] Falke and Gurevych (2017)
[13] Tauchmann, Arnold, Hanselowski, Meyer, and Mieskes (2018)
[14] Benikova, Mieskes, Meyer, and Gurevych (2016)
[15] P. V. S., Peyrard, and Meyer (2018)
[16] Zopf, Peyrard, and Eckle-Kohler (2016)
[17] Zopf (2018a)

Table 5.2.: Overview of multi-document summarization corpora. Columns are as is Table 5.1.

newswire, letters, journals, and non-fiction. For each short text, up to five reference summaries have been manually written. Each compression is at least 25% shorter than the original text.

Cohn and Lapata (2008) created a dataset that generalizes the task of sentence compression to sentence summarization. The difference is that in sentence summarization, operations such as reordering, insertion, and substitutions are allowed, whereas the focus of sentence compression is to delete words. They collected 575 sentences from 30 newspaper articles which are part of the British National Corpus and the American News Text Corpus. For each sentence, a compressed version has been created manually.

5.2.2 Headline Generation

Headline generation deals with the problem of generating a short headline for a given text. Newswire data is the most prominent text genre considered in this task. Headline generation is related to (sentence) summarization but should be considered to be a different task due to the different purposes of headlines and summaries. For example, headlines might be written in a clickbait-style. Clickbait is usually not considered to be a good summary since it does often not contain the most important information but is designed to entice a user to click on the corresponding news article. Furthermore, headlines usually contain only a few words and are not necessarily grammatically correct sentences. Filippova and Altun (2013) collect headline-sentence pairs from English news articles from the Internet. They describe that headlines are indeed very different from normal sentences that appear in longer summaries or texts. They report that headlines may not contain a main verb, omit determiners, and appear incomplete. They also propose a procedure to find a proper extractive compression of the sentences.

The English Gigaword Corpus in its fifth edition (Parker, Graff, Kong, Chen, & Maeda, 2011) contains in total 9.876.086 news articles from seven distinct sources of English newswire including the Associated Press Worldstream, the Washington Post Newswire Service, and the New York Times Newswire service. The articles cover the period from 1994 to 2010. All articles have been tokenized, and sentence segmented, named entities have been annotated, and in-document co-reference chains have been added by Napoles, Gormley, and Durme (2012). However, no summaries for the articles are available. Therefore, the corpus has mainly been used for the task of headline generation. Another application of the corpus is sentence summarization. To this end, the headlines were assumed to be summaries of the first sentence of each article (Chopra et al., 2016; Rush et al., 2015).

Colmenares, Litvak, and Sheva (2015) use a dataset of 1.3 million financial news articles for headline generation. The data has been fetched automatically from the web and contains only English articles. All articles have been published in the second half of 2012.

5.2.3 Single-Document Summarization

In single-document summarization, only one text has to be summarized. Created corpora contain therefore only one source document in the input for each input-output pair. For some manually created corpora, multiple acceptable outputs (i.e., reference summaries) have been generated to allow a better evaluation. Multiple reference summaries are usually not available in the larger, automatically crawled datasets.

Document Understanding Conference Datasets

For a long time, the datasets created for the Document Understanding Conference[3] (DUC) and the Text Analysis Conference[4] (TAC) have been used for single- and multi-document summarization. The Document Understanding Conference was first held in 2001. The topics for the 2001 dataset represent different types such as single events, multiple events with the same type, a single subject, natural disasters, biographical, and opinion (Over, 2001). For each source document, a 100-word summary was written. In the 2002 dataset, the newswire documents belong to the types natural disaster, single event, multiple events of the same type, or biography (Over & Liggett, 2002). Similarly to the 2001 dataset, two summaries have been written for each of the source documents. The DUC-2003 dataset contains very short single-document summaries with a length of only ten words (Over & Yen, 2003). The single-document summaries in DUC-2003 are generic summaries without any specific information needs of the user in mind. The DUC-2004 corpus (Over & Yen, 2004) contains for every source document in each of the 50 topics four very short (75 bytes) single-document summaries. In addition to the English summaries, the DUC-2004 corpus also contains single-document summaries for source document from 24 original Arabic news articles which have been translated to English before with a length of 75 bytes. Over, Dang, and Harman (2007) provide a detailed overview of the Document Understanding Conferences held from 2001 to 2006.

The New York Times Annotated Corpus

The New York Times (NYT) Annotated Corpus (Sandhaus, 2008) is a large corpus containing nearly every article published in The New York Times between 1987 and 2007. In total, the corpus contains 1.8 million news articles. For a subset of 650.000 articles, single-document summaries have been written by the New York Times Indexing Service transforming this large set of news articles to a large set of article-summary pairs. The corpus has been used recently to develop summarization systems (Durrett, Berkeley, Berg-kirkpatrick, Klein, & Berkeley, 2016; Hong & Nenkova, 2014; Paulus, Xiong, & Socher, 2018).

CNN/Daily Mail Dataset

Another large single-document summarization dataset has been constructed based on the CNN/Daily Mail question answering dataset (Hermann, Kocisky, & Grefenstette, 2015). Each news article in the dataset is associated with bullet points that summarize the content contained in the article. Summaries in the CNN/Daily Mail dataset are therefore not consistently written texts but rather a set of individual

[3] https://duc.nist.gov

[4] https://tac.nist.gov

sentences. To create a summary, the individual bullet points are usually just concatenated, and a potential lack of coherence is not further discussed. Due to its size, the dataset has recently also been used as training and evaluation datasets for extractive and abstractive summarization (Nallapati et al., 2017; See et al., 2017; Zopf, Loza Mencía, & Fürnkranz, 2016a).

Newsroom Dataset

The Newsroom dataset (Grusky, Naaman, & Artzi, 2018) contains single-document newswire summaries in the period from 1998 to 2017. There are two notable differences to previously published, large single-document summarization newswire datasets. First, the topics stem from 38 different sources and not only a single or few sources such as the New York Times or the CNN/Daily Mail datasets. The list of sources includes web pages such as *foxsports.com*, *time.com*, and *thesun.co.uk*. Hence, the articles have been written by many different authors, which improves the heterogeneity of the dataset with respect to writing style. Furthermore, the different sources target different audiences, which leads to broader coverage of various summary styles. Second, the summaries have been crawled automatically from web pages which contain the news articles. However, the dataset creators did, for example, not use the visible text included in a bullet point summary, but extracted summaries from the HTML metadata. Such metadata entries, often indicated by metadata field tags such as *og:description*, *twitter:description*, and *description*, are used by search engines and for distribution in social media such as Twitter or Facebook. This crawling strategy may have the disadvantage that the crawled summaries may have low quality. It is possible, for example, that the snippets have been written in a style that encourages social media users to click on the link, which leads to the full article on the original web page. Having this kind of snippets is interesting for newswire web pages since it increases the page hits and therefore the popularity and the amount of advertisement which can be presented to the readers. Similarly, teasers for news articles have a similar purpose, namely to encourage web site visitors to click on the news articles.

Large-scale Chinese Short Text Summarization (LCSTS)

The LCSTS dataset (B. Hu, Chen, & Zhu, 2015) is a large Chinese single-document summarization dataset that contains more than 2.4 million article-summary pairs. Chinese datasets are rare compared to English corpora. The articles have been crawled from Sina Weibo, a Chinese microblogging web site. Different news agencies and organizations from different domains such as politics, economics, movies, and games post articles on this blog. Only articles from popular and verified posters are included in the dataset.

5.2.4 Multi-Document Summarization

Multi-document summarization corpora differ from to single-document summarization corpora because they contain multiple source documents which have to be summarized jointly.

Document Understanding Conference Datasets

The already discussed DUC-2001 dataset does contain not only single-document summaries but also multi-document summaries. For each set of source documents, summaries with a length of 400, 200, 100, and 50 words have been written by NIST information analysts (Over, 2001). Similarly, summaries with a length of 200, 100, 50, and 10 words have been written for the DUC-2002 workshop (Over & Liggett, 2002). However, this time, two versions of each summary have been created. In addition to the abstracts, two 400 and two 200 word extracts have been created. The DUC-2003 dataset contains 100 word summaries for each topic (Over & Yen, 2003). Along with the source documents' topic, a viewpoint description, or a question was provided, which has to be used to create a proper summary. The multi-document summaries are therefore not generic in the DUC-2003 corpus but satisfy a particular information need of a reader. DUC-2004 (Over & Yen, 2004) contains four short (665 bytes) generic multi-document summaries for each topic. Furthermore, DUC-2004 contains 24 into English translated Arabic topics. DUC-2005 contains question-focused multi-document summaries for 50 topics with a length of 250 words (Dang, 2005). The questions pose a difficult information retrieval problem that cannot be answered by just returning a single name or number. An exemplary question contained in the DUC-2005 corpus is: „In the early 1990s, American tobacco companies tried to expand their business overseas. What did these companies do or try to do and where? How did their parent companies fare?“ Similarly to the DUC-2005 dataset, DUC-2006 contains 50 topics, each of which contains 25 news documents retrieved from Associated Press, New York Times, and Xinhua News Agency. Similarly, DUC-2007 contains 45 topics with 250-word multi-document summaries. Furthermore, DUC-2007 contains update summaries for the first time. The update summarization task simulates the situation that a reader is already aware of some of the documents. The 100-word update summaries do therefore not contain information retrieved from the set of already read documents but only from a set of new documents. The update summarization setup uses the same 25 documents as the non-update summarization task. In 2008, DUC became a summarization track in the Text Analysis Conference.

Text Analysis Conference Datasets

The dataset generated for the 2008 Text Analysis Conference (TAC) contains 48 topics (Dang & Owczarzak, 2008). Each topic contains two sets of source documents, each of which contains ten source documents. For the first set (set A), four query-focused summaries with a length of 100 words have been written by NIST assessors. The query describes the reader's need for information. Generic detection of important information is therefore not required in this task and can be better understood as an information retrieval task. For the second set of documents (set B), four summaries have been written under the assumption that the reader already knows the information contained in set A. This scenario models the situation that a reader got some information about a topic at some point in time and comes back to the topic a few days or weeks later to get an update for his or her information need. Instead of getting all the information contained in set A again, the reader only wants to read new information which the reader does not already know. Hence, the focus of the second summarization task is to detect redundant information in document set B with respect to document set A. Again, the summary is supposed to address the reader's information needs described by the short narrative which has already been used to create the summary for document set A. Similarly to the 2008 dataset, the 2009 TAC dataset (Dang & Owczarzak, 2009) contains 44 topics with a total of 20 source documents divided into 2 sets for query summarization and query+update summarization.

The dataset for TAC 2010 contains 46 topics (Owczarzak & Dang, 2010). Similarly to TAC 2008 and TAC 2009 datasets (Dang & Owczarzak, 2008, 2009), the source documents in TAC-2010 are divided into two sets of 10 documents each. Instead of a narrative describing the reader's information need, a set of aspects has been used to generate the summaries for TAC 2010. The aspects describe which kind of information a reader wants to see in the summary. For a natural disaster, for example, aspects could be „what happened?", „when did it happen?", „how many people have been injured?", and „which countermeasures have been taken?". Hence, the aspect-oriented summarization can be viewed as even more oriented towards information retrieval compared to question-oriented or generic summarization. For TAC 2011 (Owczarzak & Dang, 2011), a similar corpus with aspect-oriented summaries for 44 topics has been created.

Live Blog Corpus

The source documents in the Live Blog Corpus[5] are text snippets extracted from BBC and The Guardian live blogs. Live blogs are news feeds that are updated by journalists as soon as there is new information regarding the topics of the live blog available. The text snippets are rather short (62 words on average per snippet for BBC and 108 on average for the Guardian) compared to other MDS corpora.

5.2.5 Special Text Genres

The source documents of the previously discussed corpora belong to the newswire genre. We discuss corpora that contain source documents from other text genres below.

DBS

The DBS corpus[6] is recently presented MDS corpus which contains source documents and coherent extracts from 10 topics belonging to the educational domain (Benikova et al., 2016). Coherent extracts are summaries created mainly by extracting text from the source documents with a minimal amount of post-editing to produce coherent texts which are still very close to the source documents. The dataset contains source documents from different text genres such as interviews, book reviews, teaching material, NGO profiles, newspaper, and scientific articles. The dataset contains German source documents, which distinguishes DBS from many other corpora which usually contain only English sources.

[5] https://github.com/UKPLab/lrec2018-live-blog-corpus

[6] https://github.com/AIPHES/DBS

Scientific Documents

The TAC 2014 Biomedical Summarization Track[7] found that most summarization work has been done for newswire articles and focused on scientific biomedical texts instead. The corpus consists of 50 topics, each of which contains one scientific paper that belongs to the biomedical domain. For every reference paper, a set of 10 citing papers were identified which cite the reference paper. In each citing paper, the text span that most accurately reflects the citation has been annotated. The granularities of the annotations are sentence fragments, full sentences, or several consecutive sentences. All annotated sentences are considered to be a community-based summary of the reference paper. The annotations have been clustered according to the aspect that is targeted with the citation. Aspects are, for example, the goal of the paper, used methods or datasets, and conclusions. The goal of a summarization system is to identify all text spans that belong to a citation and to cluster all citation text spans according to the aspect. Furthermore, a human-written summary with a length of 250 words has been generated based on the annotated text spans and the abstract of the reference paper. Detailed information about the annotation process used to create the dataset is available online.[8]

The PubMed and the arXiv datasets[9] are large, automatically created single-document summary datasets (Cohan et al., 2018). The source document in each topic is an automatically retrieved scientific paper. The abstract of each paper in these datasets is considered to be a summary of the full paper.

User Reviews

The Opinosis[10] contains reviews about hotels, cards, and various other produced (Ganesan, Zhai, & Han, 2010). The reviews have been collected from Tripadvisor, Amazon, and Edmunds. Human annotators constructed opinion seeking queries which consist of an entity name and a topic. For example, an entity could be *Amazon Kindle*, and the topic could be *buttons*. 51 queries have been created. For each query, all sentences (approximately 100) in the reviews containing the topic for a given entity have been collected. Human annotators on Amazon Mechanical Turk created five summaries for each set of sentences. Summaries with poor quality were removed from the dataset by experts. The summary is supposed to contain the major opinions in the review set. For example, major opinions about the buttons of the Amazon Kindle could be that the buttons are too small or have a nice color. The summaries are therefore not generic but opinion-focused.

5.2.6 Structured Summaries

Besides producing textual summaries, other structures such as maps and trees can be produced and be used as summaries. Strictly speaking, this form of summarization is not covered by Definition 13 (see Section 3) since the source documents and summaries are not the same kinds of medium. It is therefore

[7] https://tac.nist.gov/2014/BiomedSumm/index.html
[8] https://tac.nist.gov/2014/BiomedSumm/guidelines/SciSumm-annotation-guidelines-V1.0.pdf
[9] https://github.com/acohan/long-summarization
[10] http://sifaka.cs.uiuc.edu/ir/downloads.html

not easily possible to compare the lengths of the two media. Hence, it is unclear if an output such as a map or a tree is indeed a reduction of the size of the input texts and therefore a summary according to Definition 13.

Concept Maps

Falke and Gurevych (2017) proposed the task of *concept map-based multi-document summarization* and created a corpus for this task. As described before, the main difference compared to traditional summarization datasets is that no textual summary has been created. Instead, the authors created a concept map, which is a directed graph. Concepts (e.g., entities, abstract ideas, events, or activities) are represented as nodes in the graph. Relationships between concepts are modeled with edges. The corpus contains 30 topics, each of which contains approximately 40 source documents and one concept map. The extraction of concepts and relationships, which form propositions, has been performed automatically using the Open Information extraction paradigm (Banko, Cafarella, Soderland, Broadhead, & Etzioni, 2007). The annotation of proposition importance has been performed with Amazon Mechanical Turk for which the authors spend $4425.45. The source documents belong to different text genres such as news articles, scientific papers, and professional blog posts, which can be considered to have a high quality, but also low-quality sources such as forum discussions, user-provided comments and personal blog posts.

Hierarchical Summarization

Tauchmann, Arnold, Hanselowski, Meyer, and Mieskes (2018) present a hierarchical summarization corpus. Summaries in this corpus are tree structures in which the root nodes of the trees represent the most general information. In the leaf nodes, the most specific information is presented. The corpus contains ten topics with source documents retrieved from the ClueWeb12 corpus (Habernal et al., 2016). For each topic, three hierarchical summaries have been generated. The corpus contains heterogeneous source documents stemming from difference text genres, including scientific articles, blogs, and forum posts. They used crowdsourcing to select important information first and asked experts to organize the selected sentences in a tree structure in a second step.

5.2.7 Limitations

Based on the comprehensive analysis of available text summarization corpora provided in the previous section, we now analyze the properties in detail to identify limitations in the current corpus landscape. For most of the properties discussed in this section, corresponding columns with more details can be found the in corpus overview provided in Table 5.1 and Table 5.2.

Limited Number of Topics

The first limitation is the lack of large multi-document summarization corpora that contain many different topics. Table 5.2 shows that multi-document summarization corpora are rather small with respect to the number of topics. The size of the usually used DUC and TAC corpora ranges from 44 in the TAC 2009 corpus (Dang & Owczarzak, 2009) to 60 topics provided by the DUC 2001 and DUC 2002 corpora (Over, 2001; Over & Liggett, 2002). The only exception among the multi-document summarization corpora with respect to the number of topics is the Live Blog corpus (P. V. S., Peyrard, & Meyer, 2018) with 2,655 topics. In the Live Blog corpus, however, the length of the source documents is rather small with an average length of approximately 90 words per source document. Usually, the source documents in multi-document summarization are much longer.

On the other hand, it can be observed in Table 5.1 that single-document summarization datasets are much larger with respect to the number of topics. The sizes range from tens of thousands of topics (Hermann et al., 2015) to more than a million topics in the largest single-document summarization corpora (Grusky et al., 2018; B. Hu et al., 2015; Sandhaus, 2008). The explanation for the availability of large single-document summarization datasets and the lack of large multi-document summarization datasets is rather simple. All large SDS datasets have been crawled automatically without the need to ask humans to write proper summaries (Grusky et al., 2018; Hermann et al., 2015). The data to generate large SDS datasets is already available on the web and can be crawled easily without requiring human effort. Grusky et al. (2018), for example, used text snippets contained in the HTML code which have been written by the authors of the articles as summaries. Summaries for MDS datasets, however, have been manually written by humans to create MDS datasets since it is much harder to find summaries for a previously selected set of documents automatically.

The lack of large datasets limits research in summarization because of two reasons. First, large datasets are crucial for a reliable assessment of the performance of summarization systems. Drawing reliable conclusions about the performance of summarization systems is difficult if test datasets are small. This problem is even more critical if the evaluation methods used (e.g., automatic evaluation with ROUGE) are not very accurate. Second, large datasets are also needed to train summarization systems. With only a few topics, it is barely possible to learn to distinguish important and unimportant information. This observation is most relevant in situations in which heuristics such as information centrality or sentence position should not be used to build more robust summarization systems. Larger training datasets can also facilitate the development of new kinds of summarization systems. The availability of the CNN / Daily Mail corpus, for example, fostered the development of abstractive summarization systems (Cheng & Lapata, 2016; Nallapati et al., 2017; See et al., 2017). This development would not have been possible with small datasets such as the DUC and TAC datasets.

High Cost for Dataset Creation

Closely related to the size of summarization datasets is the cost and effort it takes to produce datasets. One could argue that large datasets would already be available if the production was cheap and straightforward. Large datasets for single-document summarization are available since they can be easily and cheaply crawled. Unfortunately, creating multi-document datasets is a difficult task, even for humans, and therefore time consuming and expensive. Dang (2005) reports that summarizing the document sets for the DUC 2005 shared task was a difficult task: the creation of each 250-word summary consumed approximately 5 hours.

The use of crowdsourcing is an approach to reduce the cost for creating annotated data, and it has been increasingly used in NLP for a range of different tasks. However, it has been shown that the use of crowdsourcing leads to poor results for multi-document summarization (Lloret, Plaza, & Aker, 2013). We also observed in our experiments that it is difficult to get good annotation quality on platforms such as Mechanical Turk for more straightforward tasks than writing complete summaries. Hence, the usefulness of crowdsourcing to produce large summarization datasets is limited. Falke and Gurevych (2017) report that they spent approximately $4,500 for Amazon Mechanical Turk to create concept maps for concept map summarization. The annotation process resulted in 30 topics, meaning that the generation of each topic costs $150. With a cost of $150 per topic, it is a costly effort to create large summarization datasets.

Source Document Homogeneity

Another major limitation of currently available summarization datasets is the high homogeneity of the source documents. Source documents in single-document summarization datasets usually contain source documents from the newswire text genre. Scientific documents and Wikipedia pages also have been used to create single document summarization datasets (Cohan et al., 2018; Giannakopoulos et al., 2015). Covering different genres in one topic is not possible in SDS since there is only one source document per topic.

However, also in multi-document summarization datasets, we usually find only news articles in the source document sets. The DUC and TAC datasets, for example, only contain news articles from a few newspaper agencies. This narrow focus prevents the development of new summarization systems since no data is available to train and test the models for non-newswire summarization. However, learning to summarize heterogeneous source documents poses interesting new challenges. Summarizing heterogeneous corpora which do not have specific properties that can be exploited to estimate information importance easily require a deeper text understanding. A similar development can be observed in the question answering community where datasets such as the Stanford Question Answering Dataset (SQuAD) (Rajpurkar, Zhang, Lopyrev, & Liang, 2016) are criticized since good performance is possible by leveraging simple tricks which do not require any sophisticated natural language understanding (Agrawal, Batra, & Parikh, 2016; Jia & Liang, 2017).

Recently, more datasets have been created which contain more heterogeneous source documents (Benikova et al., 2016; Falke & Gurevych, 2017; Tauchmann et al., 2018). However, the available heterogeneous datasets are very small with a size of only 10 to 30 topics.

Monolingualism

Most datasets only contain source documents written in one language. English is the most dominant language used in all variants of summarization. Very few datasets are available whose source documents are written in a different language. LCSTS (B. Hu et al., 2015) is one example of a single-document summarization dataset written in Chinese and DBS (Benikova et al., 2016) is a small summarization corpus which contains German source documents. Datasets that contain many different languages are rare. The datasets used for the MultiLing shared tasks[11] are different with respect to this property and contain source documents in approximately 40 different languages (Giannakopoulos et al., 2017; Giannakopoulos et al., 2015).

Length of Summaries

The length of the summaries is usually fixed to a specific length. For example, reference summaries in the DUC and TAC datasets usually have a length of 100 words. In earlier shared tasks, summaries with a length of 50, 200, and 400 words have to be generated (Dernoncourt et al., 2018). Hence, the systems have a fixed goal for which they can be trained. However, it can be argued that for some topics, a good summary containing the most important information can be relatively long whereas for other topics a rather short summary might be sufficient to include all relevant points. Furthermore, the datasets do not model a more natural summarization situation where the length of the reference summary or summaries is not known in advance. A trade-off between precision and recall does not have to be learned in currently available datasets. Instead, the systems have to identify the most important information and add it to the summary as long as the desired length of the summary is not reached yet. Hence, a summarization dataset with varying summary lengths would be interesting to add more challenges for summarization system developers.

Length of Source Documents

The length of the source documents is, similar to the length of the reference summaries, another aspect which is usually not very large and does not have a high variance in commonly used datasets. Datasets with shorter source documents are easier to process than datasets with longer documents. This circumstance is in particular critical for summarization systems with a high computational cost. The search space for ILP-based methods, for example, increases exponentially with the number of sentences and graph-based systems or systems which have to compute similarities between all sentence pairs cannot be applied efficiently if the number of sentences is high due to the quadratically increasing costs. Furthermore, it is much harder for abstractive summarization systems to encode the source documents and use a decoder to output a summary from scratch if the source documents are long and a lot of information is encoded in the source documents.

[11] http://multiling.iit.demokritos.gr

Distribution of Content

Summarization datasets with source documents stemming from the newswire genre provide information for a specific topic. We have already discussed in Chapter 4 that journalists tend to include more important information more frequently into their articles. This observation gives rise to the assumption that a lot of overlapping content is contained in newswire source documents. Indeed, a detailed analysis shows that the omission of some source documents does not result in a substantial drop in the best possible outcome (Zopf, Peyrard, & Eckle-Kohler, 2016). If only 50% of the source documents in the DUC 2004 corpus are considered for summarization, it is still possible to achieve 100% of the optimal ROUGE$_1$ and 94% of the optimal ROUGE$_2$ score which can be achieved if all documents are considered. If only one source document is considered, 83.7% and 50.0% of the optimal scores for ROUGE$_1$ and ROUGE$_2$ can be achieved, respectively. The results indicate that the content in the source documents is redundant. The availability of redundant information is a plausible observation that is used by many summarization systems that select the most redundant information during summary creation. A more challenging dataset would distribute important information to different source documents and different locations.

Conclusions

Large single-document summarization datasets can be generated easily and cheaply since they can be crawled automatically. Creating summaries for MDS datasets is either expensive if performed by human expert annotators or has a low quality if performed with crowdsourcing. Prominent MDS corpora contain only homogeneous source documents from the newswire genre. Recently presented corpora such as the Concept Maps, the DBS, and the Hierarchical Summarization corpora approach this gap. Unfortunately, the generation of the corpora is rather expensive, and the generated corpora are even smaller than the DUC and TAC corpora. Furthermore, summarization corpora usually contain only English source documents with a few exceptions for Chinese and German. Table 5.3 summarizes the properties and limitations for popular datasets.

5.3 A New Corpus Construction Approach

We have discussed several limitations of currently available datasets in Section 5.2.7. Based on this discussion, we conclude that we would ideally like to generate large, heterogeneous multi-document summarization corpora in different languages cheaply. The crucial problem for creating such corpora cheaply is the fact that there has been no way found yet to crawl large MDS corpora automatically. If there was a way to create such datasets automatically, it might be possible to create large datasets similarly to the automatically generated large SDS datasets. In this section, we present a method to produce multi-document summarization datasets semi-automatically. To this end, we first briefly review how MDS corpora have been created in the past (5.3.1), discuss limitations (Section 5.3.2), present our new approach afterwards (Section 5.3.3), and discuss it in detail (Section 5.3.4).

Dataset	Large?	Cheap?	Multi-doc.?	Heterog.?	Multiling.?	Var. Ref. Size?	Var. Src. Size?	Distr.?
arXiv	✓	✓	✗	✗	✗	?	?	–
NYT [2]	✓	✓	✗	✗	✗	?	?	–
CNN / Daily Mail [3]	✓	✓	✗	✗	✗	?	?	–
Newsroom [4]	✓	✓	✗	✗	✗	?	?	–
DUC 2001-2007	✗	✗	✓	✗	✗	✗	✗	✗
TAC 2008-2011	✗	✗	✓	✗	✗	✗	✗	✗
TAC 2014	✗	✗	✓	✗	✗	✗	✗	?
Concept Maps [5]	✗	✗	✓	✓	✗	–	–	?
Hier. Sum [6]	✗	✗	✓	✓	✗	–	–	?
DBS [7]	✗	✗	✓	✓	✗	✓	✓	?
Live Blog [8]	✓	✓	✓	✗	✗	?	?	?
*h*MDS [9]	✗	✓	✓	✓	✗	✓	✓	✓
auto-*h*MDS [10]	✓	✓	✓	✓	✓	✓	✓	?

[1] Cohan et al. (2018)
[2] Sandhaus (2008)
[3] Hermann, Kocisky, and Grefenstette (2015)
[4] Grusky, Naaman, and Artzi (2018)
[5] Falke and Gurevych (2017)
[6] Tauchmann, Arnold, Hanselowski, Meyer, and Mieskes (2018)
[7] Benikova, Mieskes, Meyer, and Gurevych (2016)
[8] P. V. S., Peyrard, and Meyer (2018)
[9] Zopf, Peyrard, and Eckle-Kohler (2016)
[10] Zopf (2018a)

Table 5.3.: Overview of properties for popular summarization corpora with important corpus characteristics. For each corpus, we list if the corpus can be considered to be large, if its creation was cheap, if it contains multiple source documents, if it contains multilingual topics, if it contains reference documents and/or input documents with varying size, and if the important content is distributed across many input documents.

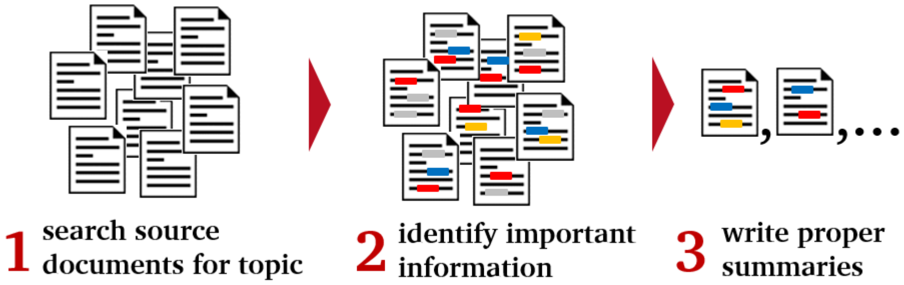

1 search source documents for topic
2 identify important information
3 write proper summaries
,,...

Figure 5.3.: Illustration of the traditional multi-document summarization corpus construction workflow. Annotators have to come up with topics and search for source documents (step 1). Important information nuggets have to identified (step 2) before high-quality summaries are written (step 3).

5.3.1 Traditional Corpus Construction Approach

Traditionally, multi-document summarization datasets are created as illustrated in Figure 5.3. First, humans select a topic and search for appropriate source documents. For the DUC/TAC corpora, for example, the topics have been chosen by NIST information analysts (Over, 2001) according to their personal interest. Choosing topics and source documents are actions that have to be executed jointly to achieve good results since humans can only pick topics for which sufficiently many source documents are available. If it turns out that there are no or not enough appropriate source documents available (e.g., because the topic is too specific) the annotators have to find other topics to create the datasets.

If a topic has been selected and enough useful source documents have been found, the annotator has to read all source documents carefully to understand all the contained information in a second step. It might be difficult for humans to get a good, comprehensive overview of a topic if there is a lot of content in the source documents. For example, the annotator might have to highlight important text parts or create mind maps to get a good overview of the topics.

After all the important information pieces have been identified, one or multiple annotators have to create well-written summaries in a third and last step.

5.3.2 Limitations

The traditional corpus construction approach has several disadvantages, which are discussed in the following. First, the annotators have to select topics on their own, which introduces a selection bias to the dataset. The topics reflect the interest of the annotators and are not necessarily representative es for a true application case. Furthermore, it requires effort and creativity to come up with topics for which enough source documents can be found. Third, the search for source documents requires domain expertise. Without a good knowledge of the topic, it is barely possible to find appropriate source documents since the annotations do not know which information is relevant for a topic. Without having domain knowledge for the topic "2016 U.S. presidential election", for example, it is impossible to know that a

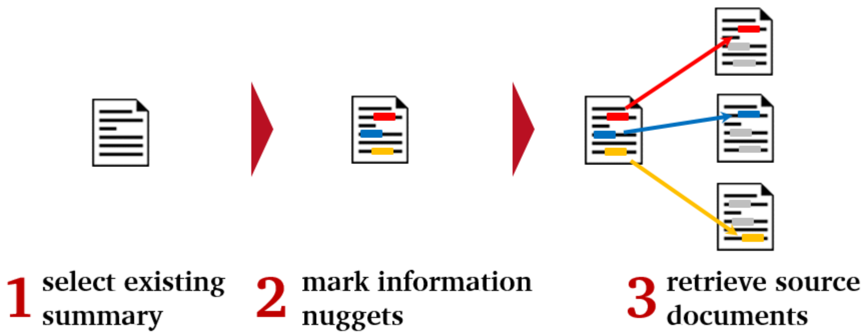

1 select existing summary
2 mark information nuggets
3 retrieve source documents

Figure 5.4.: Illustration of the new multi-document summarization corpus construction approach. Summaries are selected (step 1) and contained information nuggets are marked (step 2) before source documents are retrieved (step 3).

good set of source documents will cover subtopics such as "Donal Trump", "Hillary Clinton's private email address", and "Russian interference". Domain knowledge is not only required to compile a good set of source documents but also to identify the most important aspects of a topic. Finding, organizing, and extracting the most important information also takes a lot of time. Creating high-quality summaries is furthermore a difficult task and again requires a lot of time and expertise. As discussed before, crowdsourcing may lead to poor results (Lloret et al., 2013). Consequently, the summary generation has to be performed by experts. Another issue with this approach is that the created summaries only reflect the opinion of a single author. However, studies have shown that different annotators also create different summaries (P. V. S. & Meyer, 2017). Hence, different people consider the same information nugget differently important. This observation also aligns with the provided definition of importance in Section 2.4 in which we defined information importance with respect to a specific user or user group.

5.3.3 A New Corpus Construction Approach

We now present our new approach to generate multi-document summarization datasets, which remedies some of the issues with the traditional corpus construction approach. The key idea of the new multi-document summarization dataset is to reverse the traditional summarization strategy. Instead of writing summaries for analyzed texts, we propose to find texts that can be considered to be summaries and search for appropriate source documents. The new approach is illustrated in Figure 5.4.

In the first step, a text has to be found by a human which can be considered to be a summary. All texts which contain the most important information about a topic can be considered to be summaries. In single-document summarization, for example for scientific papers, this is rather simple since an abstract of a paper can reasonably be considered to be a summary for the paper. However, it is also possible to find texts which can be considered to be summaries not only for one single document but also for a set of documents. Survey papers, for example, can be viewed as summaries for a particular topic or recent advances in a topic. In this setup, the cited papers would be considered to be appropriate source documents for the topic. Similarly, Wikipedia pages can be considered to be summaries for the corresponding topic and the references used in the Wikipedia pages might be considered to be source documents for the topics.[12]

After a summary has been selected, the approach intends to ask human annotators to annotate atomic information pieces. Information pieces are text snippets that convey important information about the selected topic. A summary is considered to be good if it contains the information represented by the annotated text snippets. Missing important information indicates suboptimal summaries. We do not request that the text snippets have to be very meaningful without context. The text snippet *from 2006 until 2008*, for example, has been identified by our annotators to be an important text snippet for the topic *Joseph Bourdon*. Without domain knowledge, it is not possible to know what this text snippet means. However, a good summary has to mention somehow that Joseph Bourdon was a Canadian ice hockey player who played from 2006 until 2008.[13] Furthermore, we request *atomic* information pieces. We call a text snippet atomic if and only if omitting words would change the meaning of the nugget significantly (Zopf, Peyrard, & Eckle-Kohler, 2016). The atomic properties ensure that we have as little unimportant words (such as stopwords) as possible contained in the information pieces.

In a third step, source documents have to be collected which contain the annotated information pieces. If the summary already provides indicators for good source documents, they might be used for the collection, and the previously discussed text snippet annotation might be skipped. If the indicated references cannot be used as source documents, for example, because of broken or low-quality links, source documents have to be found with different strategies. One possibility is to search on the web for appropriate source documents. The information nuggets annotated in the second step can be used to find a source document for every annotated information nugget. We ask human annotators to search for diverse source documents which do not overlap much to prevent that the same information can be found in many source documents.

The summarization topics are created after the source document collection process is finished. Each topic consists of the selected summary and the retrieved source documents identically to the corpora generated with the traditional corpus collection approach.

5.3.4 Discussion

We now discuss advantages and possible issues of the new corpus construction approach.

Reduced Selection Bias

We criticized prior approaches because the annotators select topics according to their interest, which introduces a selection bias and does not result in a representative selection of summarization topics. With our approach, we can solve this issue only partially since the annotators still have to select topics.

[12] We found in our analysis that Wikipedia references are rather bad sources due to quality issues. Read more about this in Section 5.4.

[13] https://en.wikipedia.org/w/index.php?title=Luc_Bourdon&oldid=705822268

However, the availability of texts that can be used as summaries indicates that there is some general interest in this topic. Survey papers and Wikipedia articles are often written because the author thought that there is a demand for a good, updated overview of a topic. Hence, it can be argued that the selection bias is reduced due to the pre-selection of generally interesting topics by summary authors. Furthermore, it does not require creativity or effort to come up with topics as soon as a good source for potential summary texts such as all survey papers or all Wikipedia pages has been found.

Simplified Important Information Detection

Since summaries are already available, the most important information for the topics has already been identified. Hence, no domain knowledge is required to identify the most relevant information nuggets for a topic. Wikipedia articles, for example, already contain the most important information about a topic. The task of learning about a topic and identifying the most important information becomes a task where all available information in the summaries can be assumed to be important. No selection process is required anymore. Not only the need for domain knowledge but also the time which has to be spent to read all source documents is reduced, which usually consumes a substantial amount of time.

Removing personal biases in the importance estimation process is another advantage of the new corpus construction approach. The annotators do not have to decide which information is important anymore. Hence, they do not introduce personal biases into the summaries. Instead, the decisions are made by domain experts who wrote the summary text collaboratively. Consequently, the summaries do not reflect the opinion of only one annotator but the consensus of many authors, which leads to more representative summaries.

Manual Search for Source Documents

Even though the identification process of important information has been simplified, some human effort is still required to annotate information nuggets and search for source documents that contain the information nuggets if potentially available references cannot be used as pointers to source documents. The new corpus construction approach can therefore not be performed without any human effort and therefore still requires a potentially expensive component.[14]

Distribution of Content

Another result of applying the new corpus construction approach is a distribution of important content across the source documents if they are searched manually since the approach stipulates that one source document should be found for each information nugget. This property requires the summarization systems to analyze all source documents to make sure to find all important information nuggets. It becomes less likely that it is sufficient only to analyze a few source documents to create a very good summary.

[14] We discuss a strategy which removes the need for the last remaining human effort in Section 5.5.

No Summary Writing Required

The most important advantage of the new corpus construction approach is that it is not necessary to write reference summaries manually since the approach reuses already available summaries. Writing summaries consumes a lot of time (Dang, 2005) and is therefore very expensive. Furthermore, writing expertise is required to create high-quality summaries. Using already available summaries is also the key ingredient that allows creating large single-document summarization datasets.

Summary Quality Critical

Ensuring a high quality of the summaries is essential for creating a high-quality corpus with the new approach since the summaries are not written with already selected source documents in mind but are selected before the source documents are known. Two critical problems can occur. First, a prospective summary may contain unimportant information. This situation is problematic since summarization systems would be rewarded if they also included unimportant information nuggets that are present in the source documents into summaries. Second, it is possible that the author of the summary missed to include essential information into the summary. In this case, summarization systems would be not rewarded if they identified important information correctly as important. The presence of unimportant information and the absence of important information in the summary are therefore two potential problems that should be avoided by selecting high-quality summaries which contain the most important information about a topic and which do not contain unimportant information.

Source Document Quality

The quality of the source documents is compared to the quality of the summary is a rather uncritical issue for the following reasons. Similarly to the previously described issues, two situations can occur. First, important information which is present in the summary can be missing in the source documents. Even though this limits the quality of the optimal reachable output in the case of extractive summarization, it is not an issue for assessing the information importance estimation performance of the summarization model since the model is only asked to estimate the information importance of available information. If a model is able to estimate the information importance of the available information correctly, it will get the best possible quality assessment for this topic. Second, unimportant or even unrelated information can be present in the source documents. This situation is also no issue but can instead be considered to be a desirable property of the corpus. If unimportant information is present, the summarization systems have to be able to detect that this information is unimportant and should not be included in the summary.

Number of Summaries per Topic Limited

One inherent disadvantage of the new corpus construction approach is that only one reference summary per topic can be retrieved without cost. Automatic evaluation of summarization systems with the state-of-the-art evaluation tool ROUGE (C.-Y. Lin, 2004) benefits from the availability of multiple reference summaries (C.-Y. Lin, 2004). The availability of more summarization topics (i.e., a larger dataset) can mitigate this issue such that similarly reliable estimation of summarization system performance is possible even if multiple reference summaries are not available.

5.4 Semi-automatic Creation of *h*MDS

Based on the previously described new corpus construction approach, we created a new summarization dataset called *h*MDS. We aimed at achieving several additional goals (Section 5.4.1), which have been identified as missing in usually used summarization corpora as discussed in Section 5.2, in addition to a first application of the new corpus construction approach (Section 5.4.2).

5.4.1 Additional Goals

In this section, we discuss the additional goals that we want to achieve by building a new multi-document summarization corpus. These goals are independent of the new corpus construction approach. We aim additionally for a high heterogeneity, the availability of unimportant or even unrelated content, and a distribution of content across the source documents.

Heterogeneity

We have already discussed in Section 5.2.7 that standard summarization datasets usually have a very homogeneous set of source documents. A major goal of this corpus creation effort is to generate a more heterogeneous corpus to close this resource gap. In particular, we aim at creating a corpus in which simple heuristics are not sufficient to perform well, similar to the recent efforts to create question answering datasets, which cannot be solved using simple tricks. We hope that the dataset does not allow us the sentence position feature, for example, as importance indicator since important information can appear at any place in non-newswire genres such as chat logs.

Distribution of Content

Additionally, we would like to create a corpus in which the important information nuggets are distributed across all source documents which forces summarization systems to analyze all source documents. In combination with the heterogeneity property, it means that summarization systems have to be able to find important information in many different text genres in which different underlying patterns are present.

Varying Document Lengths

We also would like to have summarization datasets with different summary and source document lengths. Former is interesting since summarization models should be able to generate short and long summaries if required and they should also be able to know how long a summary should be ideally if no desired lengths are specified in advance. Source documents with varying lengths pose new challenges to create computationally efficient summarization systems.

Presence of Unimportant or Unrelated Information

We also want to create a corpus where a lot of unimportant or even unrelated information is present in the source documents. Important information should be hard to find and should be hidden in a lot of unimportant information. By adding a lot of unimportant information, the difference between good and bad summarization systems becomes clearer since poor summarization systems have many opportunities to make mistakes, which can only be avoided by good summarization systems that are able to understand the content. Only good summarization systems are able to find the needle in the haystack.

5.4.2 Creating *h*MDS

We describe in the following how we created *h*MDS on the basis of the previously proposed corpus construction approach.[15]

Using Wikipedia Featured Articles as Summaries

The quality of the prospective summaries is crucial to create a high-quality summarization corpus with the new corpus construction approach, as discussed in Section 5.3.4. We selected Wikipedia as a source for prospective summaries and used only the first section of Wikipedia featured articles (the so-called *lead* section). Wikipedia featured articles are articles with very high quality. The articles have to fulfill several quality criteria to get the *featured*-label. A featured article has to be

1. **well-written:** its prose is engaging and of a professional standard;
2. **comprehensive:** it neglects no major facts or details and places the subject in context;
3. **well-researched:** it is a thorough and representative survey of the relevant literature; claims are verifiable against high-quality, reliable sources and are supported by inline citations where appropriate;
4. **neutral:** it presents views fairly and without bias; and
5. **stable:** it is not subject to ongoing edit wars, and its content does not change significantly from day to day, except in response to the featured article process

according to the Wikipedia quality criteria for featured articles.[16] Particularly important is a further criterion regarding the lead section of featured articles which requests that every featured article has

[15] More annotation details can be found at the corresponding Github page: https://github.com/AIPHES/hMDS

[16] https://en.wikipedia.org/wiki/Wikipedia:Featured_article_criteria

to contain "a concise lead section that summarizes the topic and prepares the reader for the detail in the subsequent sections". Furthermore, it is requested that the article "stays focused on the main topic without going into unnecessary detail and uses summary style". Hence, lead sections in Wikipedia's featured articles can be considered to be high-quality texts which can be used as summaries to create a new corpus.

Furthermore, Wikipedia featured articles represent a wide range of topics from many different domains that are relevant to many users. Only topics with a reasonable amount of public interest are included in Wikipedia. Wikipedia states that a topic has to have received significant coverage in reliable sources that are independent of the subject to be suitable for a stand-alone article or list.[17] Hence, articles without public interest are deleted. The selection bias discussed previously is therefore mitigated by selecting topics from the set of Wikipedia featured articles.

Since Wikipedia articles are written by many authors, we can furthermore consider the lead of a featured article as the consensus of many people regarding the important information about a particular topic. Hence, the leads formulate a representative opinion about the importance of information instead of just reflecting the opinion of one single author. This fact makes Wikipedia featured articles even more appropriate for the intended purpose.

Choice of Topics

Since Wikipedia provides a large number of featured articles, we first selected a subset of the articles to create *h*MDS. We selected three broad domains based on the featured article overview page[18], namely

1. Art, Architecture, and Archaeology (D1);
2. History (D2); and
3. Law, Politics, and Government (D3),

and selected individual articles from these domains.

Annotating Information Nuggets

After choosing the articles, we asked three marginally trained annotators to annotate and extract roughly 10 to 20 information nuggets from the lead of each Wikipedia article. The extracted information nuggets have been stored in metadata files which are included in the corpus dataset.

It was necessary to annotate information nuggets since we found that the quality of the referenced pages is not sufficient to retrieve high-quality source documents. We found issues such as broken links, outdated web pages, or pages where no information about the claimed statement could be found. Nev-

[17] https://en.wikipedia.org/wiki/Wikipedia:Notability

[18] https://en.wikipedia.org/wiki/Wikipedia:Featured_articles

ertheless, the references have been used to identify source documents in subsequent work (P. J. Liu et al., 2018), which is similar to our presented approach.

Finding Source Documents

Given the list of information nuggets, we asked the annotators to search for source documents that contain annotated nuggets. The annotators used web search engines such as Microsoft Bing[19] and Google Search[20] to find good source documents. The annotators used the topic title (i.e., the title of the Wikipedia page) in addition to the information nugget as search terms. We allowed annotators to skip topics if it became apparent that not enough sources can be found for a topic. To achieve the goal of creating a heterogeneous corpus, we asked the annotators to select the source documents such that many different text genres are covered. We predefined a list of 10 different text genres and asked the annotators to classify the retrieved texts accordingly.[21] An additional requirement for the retrieved source documents is that they have to be sufficiently different from the reference summary. Otherwise, it would be possible for summarization systems to return (parts of) a single source document to achieve high performance. The retrieved documents have been archived in the Wayback Machine[22] to make them available for a long period.

Since the source documents are web pages and not ready-to-use text documents, we asked the annotators to extract and store the relevant part of the web page in a text file. We also generated a version of the corpus by performing automatic boilerplate content removal with Boilerpipe (Kohlschütter, Fankhauser, & Nejdl, 2010). A third version contains all visible web page content. We use the shorthand notation *h*MDS-M, *h*MDS-A, and *h*MDS-V to denote the manually extracted, the automatically extracted, and the version with all the visible content, respectively. We also created a version of the corpus where sentence splitting has already been applied since most extractive summarizing systems extract full sentences. Providing sentence splitting improves the reproducibility of summarization experiments since it removes one noisy preprocessing step. We used version 1.7.0 of the Stanford Segmenter in the DKPro Core software (Eckart de Castilho & Gurevych, 2014) to perform sentence segmentation.

5.4.3 Analysis

In this chapter, we analyze the result of our corpus creation effort. We mainly compare to the DUC 2004 and TAC 2008 datasets since they are the most prominent multi-document summarization datasets.

[19] https://www.bing.com

[20] https://www.google.com

[21] More details regarding the text genres can be found in Section 5.4.3.

[22] http://archive.org/web

Text Genre – Description (Examples)	Count	across domains D1	D2	D3	Avg. length (in words)
article – well-written text (high-quality blog post, news article)	524	0.37	0.42	0.47	1452.70 ± 1751.28
forum post – lack text structure (QA site, Youtube comment)	115	0.10	0.08	0.09	964.10 ± 1726.89
microblog – short, contains abbreviations (Twitter)	33	0.03	0.03	0.02	53.61 ± 14.44
organization – announcement, press release (any org./company)	99	0.11	0.06	0.06	749.29 ± 1119.21
encyclopedic short – encyc. source (Urban Dictionary, IMDB)	115	0.12	0.07	0.08	400.45 ± 362.88
encyclopedic long – encyc. source (Wikipedia)	137	0.11	0.14	0.07	3434.15 ± 5077.32
social media – post in social network (Facebook, Google+)	11	0.01	0.01	0.01	270.45 ± 250.67
scientific – contain citations and bibliography	119	0.07	0.08	0.14	5394.03 ± 9118.11
education – text book, tutorial	79	0.05	0.09	0.05	1568.76 ± 3020.62
dialogues – opinionated (interview, transcript, discussion)	33	0.02	0.03	0.03	3759.79 ± 4897.97
Total	1265	0.36	0.35	0.28	1863.59 ± 3928.91

Table 5.4.: List of genres present in the *h*MDS corpus along with their fractions in the different domains. The length details are computed for the M-version of the corpus.

Corpus Size

In total, we created 91 summary-source documents pairs (i.e., topics). The annotators retrieved 1,265 source documents for these topics. Hence, the corpus is larger in terms of topics and number of source documents than the DUC and TAC corpora. We obtained 13.90 ± 3.09 source documents per topic. The varying number of source documents is another distinguishing feature of our corpus. In comparison, the DUC 2004 and TAC 2008 corpora have exactly ten source documents per topic (i.e., 10 ± 0).

Text Genres

Since we asked the annotators to classify each source document according to the text genre it belongs to, we are able to provide a detailed analysis of the distribution of texts genres in our corpus. Table 5.4 provides an overview of the text genres which are present in our corpus, as well as their distributions.

Most source documents belong to the article genre. The distribution of the other genres is less skewed with most documents belonging to the encyclopedic, scientific, and forum post categories. The fraction of microblog documents, dialogues, and social media is, however, considerably smaller.

On average, we obtained 5.39 ± 1.54 different genres per topic with a minimum of 3 and a maximum of 9 different genres per topic. The results show that the topics in the *h*MDS corpus contain sources from very diverse genres. We also observe variations of the distributions of text genres across the three different domains.

Corpus	Avg. length (in words)	Relative std
*h*MDS-M	1863.59 ± 3928.91	2.11
*h*MDS-A	2192.53 ± 8196.75	3.74
*h*MDS-V	2973.06 ± 8429.32	2.84
DUC 2004	672.14 ± 506.32	0.75
TAC 2008	589.20 ± 480.33	0.82

Table 5.5.: Length comparisons of source documents in previous corpora and *h*MDS.

Corpus	Avg. length (in words)	Relative std
*h*MDS	245.55 ± 132.94	0.54
DUC 2004	118.11 ± 6.38	0.05
TAC 2008	109.33 ± 7.01	0.06

Table 5.6.: Length comparisons of summaries in previous corpora and *h*MDS.

Length of Source Documents and Summaries

Another property that we analyze is the length of source documents and summaries. Table 5.4 provides information about the distribution of lengths across the different genres. We see that we obtained a wide variety of different lengths across the genres. Since Table 5.4 only provides information for *h*MDS-M, we provide more details of the source document lengths in Table 5.5 where we can see that the variation of lengths increases strongly in the versions A and V of the *h*MDS corpus. Compared to the DUC 2004 and TAC 2008 corpora, we obtained much longer source documents, as well as a much higher variance in length.

Regarding the summaries, we achieved a substantial difference to prior work as well (see Table 5.6). Our summaries are on average about twice as long as the summaries in DUC 2004 and TAC 2008. The major difference, however, is the large variance of lengths in our corpus, which can be observed by both standard- and relative standard deviation. The minimum length of a summary in our corpus equals 72 words, and the maximum length equals 657 words.

Textual Heterogeneity

Heterogeneous documents are expected to use different wording and to have topics shifts. In order to measure this textual heterogeneity, we use an information theoretic metric on word probability distributions. In our experiments, we use the Jensen-Shannon (JS) divergence. In contrast to the Kullback Leibler (KL) divergence (Kullback & Leibler, 1951), it is symmetric and it is always a finite value. It incorporates the idea that the distance between two distributions cannot be very different from the average of distances from their mean distribution. It is defined as

$$JS(P||Q) = \frac{1}{2}KL(P||A) + \frac{1}{2}KL(Q||A), \tag{5.1}$$

	DUC 2004	TAC 2008	*h*MDS-M	*h*MDS-A	*h*MDS-V
TH_{JS}	0.3019	0.3188	0.3815	0.3358	0.3252

Table 5.7.: Average TH_{JS} scores of classical corpora and our new datasets.

where $A = \frac{P+Q}{2}$ is the mean distribution of P and Q. Based on the JS divergence, we can define a measure of textual heterogeneity TH for a topic T composed of documents $d_1, \ldots, d_n$ as

$$TH_{JS}(T) = \frac{1}{n} \sum_{d_i \in T} JS(P_{d_i}, P_{T \setminus d_i}) \tag{5.2}$$

where P_{d_i} is the probability distribution of words in document d_i and $P_{T \setminus d_i}$ is the probability distribution of words in all other documents of the topic except d_i. TH_{JS} is the average divergence of documents with all the others and provides therefore a measure of diversity among documents of a given topic.

Table 5.7 shows the results of the analysis according to the TH metric. DUC 2004 and TAC 2008 have similar source documents in comparison to *h*MDS-M according to TH. *h*MDS-A and *h*MDS-V are closer to the classical datasets than *h*MDS-M. We observe this result because these versions contain much more boilerplate content compared to *h*MDS-M, which makes them more similar again. Nevertheless, the textual heterogeneity measure provides evidence that the main content in the source documents is more heterogeneous than in the DUC 2004 and TAC 2008 datasets.

Distribution of Content

	ROUGE$_1$ Recall				ROUGE$_2$ Recall			
Corpus	full	n-1	n/2	1	full	n-1	n/2	1
*h*MDS-M	0.68	0.67 ± 0.03	0.63 ± 0.06	0.42 ± 0.12	0.48	0.46 ± 0.04	0.40 ± 0.07	0.17 ± 0.09
DUC2004	0.43	0.43 ± 0.01	0.43 ± 0.02	0.36 ± 0.05	0.16	0.15 ± 0.01	0.15 ± 0.01	0.09 ± 0.03
TAC2008A	0.46	0.46 ± 0.02	0.44 ± 0.02	0.35 ± 0.05	0.20	0.19 ± 0.01	0.17 ± 0.02	0.10 ± 0.03

Table 5.8.: Results of the content distribution experiment. We present results of using all, n-1, n/2, and only one source document. The omitted documents were selected randomly.

As mentioned above, we aim for a distribution of content across all source documents in the corpus. To evaluate this property, we conduct an experiment in which we use different fractions of the source documents as input for an oracle greedy summarization system that has access to the reference summaries. It is therefore not a summarization system but can be used to estimate the best possible output given a set of input documents. We generated for this experiment different subsets of the full set of available sentences. The results of the experiment are displayed in Table 5.8. We report the best possible scores that can be achieved with a greedy selection strategy if the reference summaries are known for the full set of source documents, for a subset in which one source document is missing, for 50% of the source documents, and for only one source document. The documents which have been removed from the set of source documents have been removed randomly.

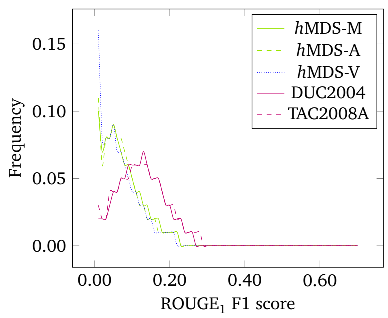


Figure 5.5.: F1 score

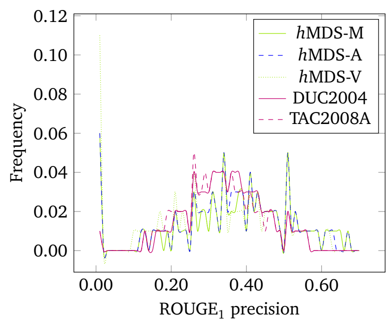


Figure 5.6.: Precision

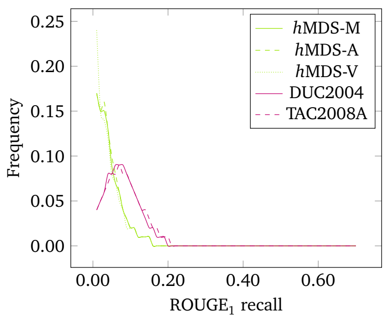


Figure 5.7.: Recall

Table 5.8 shows that the omission of one document has already an effect in our corpus. The output of the oracle greedy selection decreases by 1% point for ROUGE$_1$ recall and by 2% points for ROUGE$_2$ recall. The optimal performance differs by 1% point for ROUGE$_2$ and is constant for ROUGE$_1$ in DUC 2004 and TAC 2008. The effect increases when only 50% of the source documents are used as input for the greedy selection. A rather large difference can be observed when only one source document is available to the summarizer. In DUC 2004 and TAC 2008, the summarizer is still able to achieve 83.7% and 76.1% of the optimal score. In our corpus, only 61.8% can be achieved according to ROUGE$_1$. For ROUGE$_2$, we observe 56.3% and 50.0% in DUC 2004 and TAC 2008, compared to 35.4% in our corpus. Due to the high annotation effort, we could only investigate the differences according to ROUGE scores, which also considers stop words in the default setup, which is one explanation that the variation is not even higher.

Distribution of ROUGE Scores

In this section, we investigate the distribution of ROUGE$_1$ scores of single sentences in our corpus compared to the DUC 2004 and TAC 2008 corpora. Figures 5.5, 5.6, and 5.7 provide the distribution of ROUGE$_1$ F1 measure, ROUGE$_1$ precision, and ROUGE$_1$ recall, respectively. The evaluation shows that the distribution according to ROUGE$_1$ similar, except for a large number of sentences with very low precision in our corpora. The ROUGE$_1$ recall curve shows that single sentences in DUC 2004 and TAC 2008 on average provide a higher recall compared to *h*MDS. In combination, we see that there are many sentences in *h*MDS with both very low precision and very low recall. Thus, we can conclude that we indeed constructed a corpus containing sentences that do not contribute much to a good summary.

5.4.4 Conclusions

We created the *h*MDS corpus which is a *heterogeneous* multi-document summarization dataset. The heterogeneity poses new challenges for research in summarization since simple heuristics that work well in newswire (or other single-genre datasets) will not lead to a very good performance. A better understanding of language is required to solve *h*MDS. Furthermore, the created corpus is larger than standard multi-document summarization datasets and contains topics of general interest.

*h*MDS shows that the new corpus construction approach can be used to create multi-document summarization datasets. However, human effort is still required to apply the new corpus construction approach. We therefore further simplify the corpus construction in the Section 5.5.

Although the primary purpose of our corpus is multi-document summarization, it can also be used for other tasks. Since we store the genre of each source document, it can be used for text genre classification, or as a dataset for training and evaluating boilerplate removal systems. Research in automatic source document retrieval might also make use of the created dataset.

If the new corpus construction approach is used to create a multi-document summarization dataset, it is not necessary to ask human annotators to produce summaries for the selected source documents. Hence, we did not ask our annotators to read the source documents and to create a summary. Hence, the human performance in the *h*MDS dataset is yet unknown. It would be very interesting to know how well humans can perform in the created dataset. However, we decided to not perform this experiment due to the high additional costs.

5.5 auto-hMDS

Even though the new corpus construction approach presented in Section 5.3 does not require human effort to produce summaries, it still requires manual effort to annotate information nuggets and to search for appropriate source documents. This fact prevents a fully automatic corpus construction. Hence, creating huge summarization datasets as they are known in single-document summarization can still be expensive. In this chapter, we present a modification of the construction approach, which removes the last remaining human effort and, hence, results in a fully automatic corpus construction strategy.

5.5.1 Modifications

We describe in this section how we modify the corpus construction approach presented in Section 5.3.4 to remove the last required human effort.

Summary Collection

In the previous section, we extracted 91 English featured article leads manually and generated 91 topics based on the extracted summaries. We do not perform a manual selection of specific topics anymore in the extended version of the construction approach but perform this task automatically and retrieve the currently available lead sections of all topics in the English and German Wikipedia. After creating a list with all Wikipedia featured articles, we use the MediaWiki Action API[23] to retrieve the lead sections of 7,613 Wikipedia articles.[24] Every article is the seed for one summarization topic. We use the full lead section of each article as summary and do not truncate longer lead sections.

[23] https://www.mediawiki.org/wiki/API:Main_page

[24] The lead section can be extracted with the API request „extracts|info&exintro&explaintext“.

Finding Source Documents

Furthermore, we asked human annotators to annotate information nuggets and to use the annotated text spans as queries in web search engines to find source documents. This task consumed most of the annotation time and can be considered to be the most expensive. Instead of extracting information nuggets, we propose to directly use the sentences contained in the lead sections together with the topic name as search terms without any intermediate annotation. We found that this strategy works well since sentences in Wikipedia leads are usually rather short and often focused on one piece of information. Similar to the $hMDS$ construction, we performed sentence splitting to obtain individual sentences.

For all sentences in all Wikipedia articles, we use the Google Custom Search Engine[25] (CSE) to search for source documents. We use the topic name together with the sentence as query term for the CSE. The rights field of the CSE is configured only to find sources that can be freely used for non-commercial use-cases.

Retrieving Source Documents

The result of the invocation of the Google CSE is a link list pointing to web pages that contain the provided query terms (topic name + sentence text). For each sentence, we retrieved up to ten links. Since some of the sentences occur only rarely on web pages which can be used for non-commercial use-cases, we did not obtain ten links for each query. For each sentence, we tried to download the first web page in the query result. If a page was not available, we continued with the next URL until we were able to download a page or reached the end of the query result list. To retrieve the best possible snapshot of the web page, we did not only download the HTML code of the web page but rendered every web page using the Google Chrome browser. We used the Selenium[26] framework to interact with the browser automatically. This approach creates better snapshots of web pages since dynamic content can be created or modified (e.g., with JavaScript) before the snapshot is taken. It turned out that using a browser to retrieve the page content improves the quality of the snapshots in particular for web pages which use a lot of JavaScript, such as many blogs or Youtube.

5.5.2 Analysis

Corpus Size

In total, we created 5,132 English and 2,481 German topics. Every topic contains one reference summary file, one file which contains one sentence per line (constructed with automatic sentences splitting), and for each sentence a list of URLs. 71,162 and 22,303 sentences are contained in the English and German summaries, respectively. We found 473,754 (on average 6.66 per sentence) and 75,594 (in average 3.39 per sentence) URLs for the German and English corpora, respectively. English and German lead sections have an average length of 13.87 and 8.99 sentences.

[25] https://developers.google.com/custom-search/json-api/v1/overview

[26] http://www.seleniumhq.org

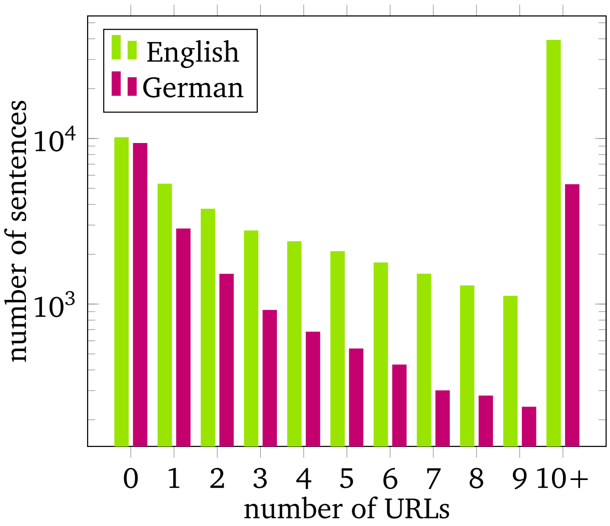


Figure 5.8.: Distribution of retrieved URLs for sentences in the auto-*h*MDS corpus. '10+' refers to the number of sentences with 10 or more URLs.

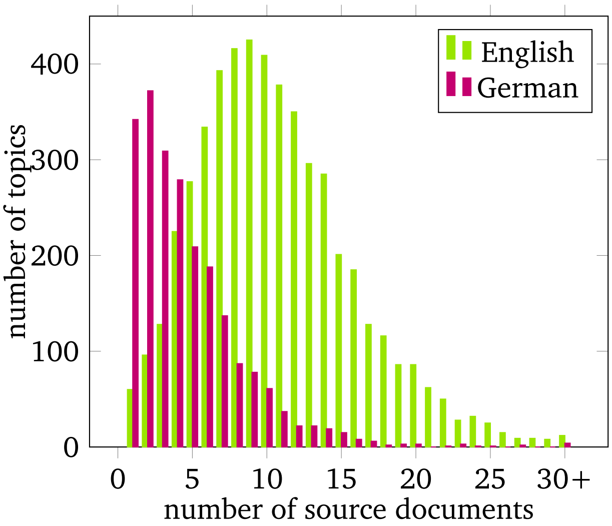


Figure 5.9.: Distribution of source documents in the auto-*h*MDS corpus. '30+' means 30 or more source documents.

Figure 5.8 shows a distribution of number of URLs per sentences. We were not able to retrieve useful source documents for a significant number of sentences with the search engine in particular for German sentences (English: 10,102 (14.20%), German: 9,325 (41.81%) sentences with an empty search results). One reason for this is the search engine configuration. We aimed at retrieving only URLs that can be freely used for non-commercial use-cases. More results might be retrieved if the search was not restricted. Important to note is that lacking source documents is no limitation of the quality of the corpus as already discussed previously. Based on the collected URLs, we tried to retrieve one source document for each sentence. We removed all the topics for which we were not able to retrieve any source documents. Finally, we obtained a corpus size of 5,106 English and 2,210 German summarization topics. Figure 5.9 shows the distribution of the number of source documents across the remaining topics.

Costs

In total, we performed about 93k searches with the Google Custom Search Engine. Since Google charged \$5 per 1000 queries, the retrieval of the links for all the sentences costs about \$500. No further expenses had to be made due to the high degree of corpus construction automation. Hence, the creation of one topic in the auto-*h*MDS corpus only costs less than \$0.07 on average. This result is fairly cheap compared to the costs of paying humans to select topics, search for source documents, and write summaries as it was performed for other multi-document summarization corpora. In total, we obtained about 550k links, in average 5.90 links per query.

5.5.3 Summarization experiments

We also performed experiments with baseline summarization methods for auto-*h*MDS to provide the first reference points for future research experiments. The *Random* baseline chooses sentences randomly until the summary reached the desired length. *Lead* uses the first sentences of the source documents. ROUGE$_2$ and ROUGE$_2$ choose the best sentences greedily according to the ROUGE$_1$ recall and ROUGE$_2$ recall score of individual sentences. To compute the scores, both summarization systems use the reference summary. Therefore, both ROUGE$_1$ and ROUGE$_2$ cannot be considered to be competitive summarization systems but are rather indicators for the best possible scores which can be achieved. The results of the experiments can be found in Table 5.9.

	100 words		200 words	
system	R-1	R-2	R-1	R-2
Random	0.1857	0.0185	0.2553	0.0325
Lead	0.1229	0.0261	0.1056	0.0228
ROUGE$_1$	0.4302	0.2161	0.4769	0.2117
ROUGE$_2$	0.4594	0.2927	0.4864	0.2924
Random	0.2290	0.0286	0.2841	0.0434
Lead	0.2524	0.0699	0.2676	0.0790
ROUGE$_1$	0.5601	0.3812	0.5168	0.3022

Table 5.9.: Summarization performance of different summarization systems in the auto-*h*MDS corpus for different summary lengths (100 and 200 words) and different ROUGE versions (ROUGE$_1$ and ROUGE$_2$) for the German (top) and the English (bottom) part of the corpus.

We observe that there is a large performance gap between the *Random* and the *Lead* baselines and the upper bounds achieved by ROUGE$_1$ and ROUGE$_2$. This result is promising since it indicates that the area between random guessing and a very good summarization system is large. Hence, the corpus will be useful to distinguish between good and bad summarization systems. Another interesting observation is that ROUGE scores for the English part are higher than in the German part of the corpus. Not only the baselines achieve higher scores but also the upper bound seems to be higher for English texts.

5.6 Summary

In this chapter, we discussed data that can be used to train and evaluate machine learning models to estimate the importance of information. We conclude that available datasets are focused on specific text genres. This focus is problematic since the individual text genres provide individual signals of importance, which can be used by summarization systems. The systems do not have to understand the idea of information importance but can rely on surrogate signals to perform well in the datasets.

Furthermore, we see that the construction of multi-document summarization datasets is expensive, which leads to few and small datasets, which is in particular problematic for data-hungry machine learning systems. It cannot be expected that a machine learning system will be able to learn the complex concept of information importance from only a few data points.

To resolve the issue with small datasets, we developed a novel corpus construction approach that allows a semi-automatic and also a fully automatic corpus construction. No more human effort is required to create huge summarization datasets.

Using this approach, we created the *h*MDS and the auto-*h*MDS datasets. *h* stands for heterogeneous and emphasizes the heterogeneous nature of the corpora. We did not only include source documents from one single text genre such as news articles, blog posts, or social media content but created a dataset that includes a wide range of text genres in every single summarization topic.

The extraction can also be performed without human effort for many other languages available on Wikipedia. Hence, the approach can easily be extended to more languages. There are also many reasonably good articles in Wikipedia which are currently not featured articles. We currently exclude these articles in our corpus construction to achieve a high-quality dataset. If one aims at creating an even larger, but possibly lower-quality dataset, it is easy to modify the query to retrieve not only the lead of featured articles but all articles which are present in Wikipedia. Furthermore, it is possible to search for other sources for already available summaries than relying only on Wikipedia.

6 Learning to Estimate Context-free Information Importance

Algorithms that are able to learn from experience are the second major ingredient required for machine learning. Hence, we develop in this chapter new machine learning models for context-free information importance estimation and discuss the next research question: **How can machines learn context-free estimation of information importance?**

We provide foundations of machine learning in Section 6.1 where we discuss how data can be used to directly or indirectly learn to perform a task.

In Section 6.2, we review prior works which are related to context-free information importance estimation. As discussed previously, context-free information importance estimation is necessary to construct reliable information importance estimators.

In Section 6.3, we propose a fully context-free information importance estimator. We present the the basic learning algorithm, which is based on learning from pairwise preferences, in Section 6.3.1. We extend the idea of pairwise preferences to *contextual* pairwise preferences in Section 6.3.2. We use contextual pairwise preferences to estimate the importance of elements in the context of already selected elements. This is important for greedy algorithms to avoid redundancy. Moreover, contextual pairwise preferences can also model synergy effects, which has not yet been investigated by prior models. In Section 6.3.3, we provide an example which illustrates how the presented model works. The presented model relies on using a basic type of element for information importance estimation. We investigate how well the model performs with bigrams, one of the simplest representations, in Section 6.3.2. In Section 6.3.5, we examine a wide range of possible annotation types. We also use the newly presented evaluation methods described in Section 3.5 for the first time. Section 6.3.6 concludes this section.

In Section 6.4, we discuss sentence regression, which uses supervision more directly than the previously presented approach. Again, most prior works rely on document-dependent importance estimation. We discuss prior works in Section 6.4.1. To allow a sound formulation of the problem, the question which regressand has to be used in the regression problem. We provide an illustration of the problem in Section 6.4.2, and show that prior works use a suboptimal regressand in Section 6.4.3. The results also imply that comparisons of prior works between greedy approaches and methods based on integer linear programming (ILP) are biased in favor of ILP methods. We conclude this section in Section 6.4.4.

Section 6.5 provides a summary of this chapter.

6.1 Foundations

We discussed tasks and datasets in Section 5.1, which are the basis for machine learning. We recall that a task can be described by a function f which specifies the desired output for different inputs. Machine

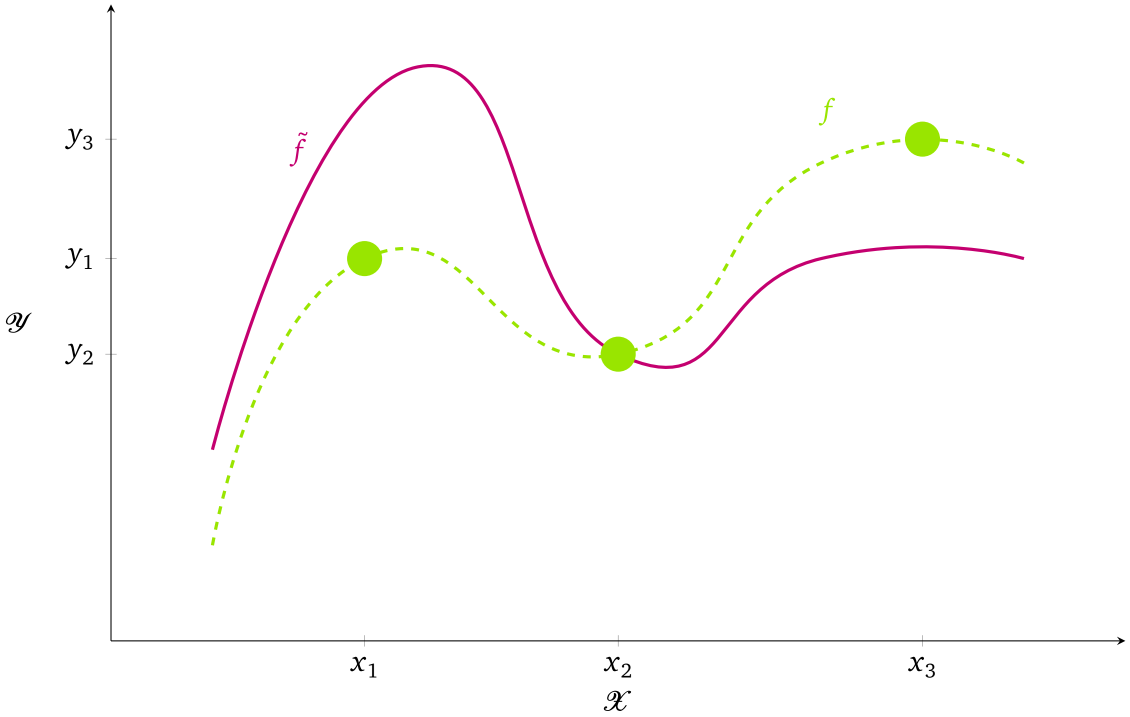


Figure 6.1.: Illustration of learning models in machine learning. The visualization extends Figure 5.2 by adding a second function $\tilde{f}$ which aims at approximating the given function f.

learning datasets $\mathbf{D} = (\mathbf{X}, \mathbf{Y})$ usually contain only a subset of possible input-output pairs due to the size of domain and codomain of tasks. Individual input-output pairs are denoted by $(\mathbf{X}_i, \mathbf{Y}_i)$ such that $\mathbf{Y}_i = f(\mathbf{X}_i)$.

For some tasks, it is rather simple to write algorithms that are able to compute the desired output for a given input. Computing square-roots for real-valued numbers, for example, can be performed rather quickly. For many interesting tasks, however, it is relatively difficult to write an algorithm directly. Writing an algorithm that is able to translate sentences or which is able to detect faces in images turns out to be very difficult.

Supervised Learning

A solution approach for these kinds of tasks is *supervised machine learning*. Supervised machine learning aims at *training* machines to perform a task based on experience instead of writing a computer algorithm that performs the task directly. We formalize supervised machine learning in the following. To this end, we first extend the terminology introduced in Section 5.1 by adding a second function $\tilde{f}$. $\tilde{f}$ represents the function which is learned by a machine learning algorithm. Informally speaking, $\tilde{f}$ represents the conception of the anticipated task f learned by the machine learning algorithm. We illustrate $\tilde{f}$ in Figure 6.1 in addition to the anticipated task f which is represented by the dataset $\mathbf{D} = (\mathbf{X}, \mathbf{Y})$.

Supervised machine learning aims at finding a function f which approximates f as good as possible. The experience provided in supervised learning are pairs $(\mathbf{X}_i, \mathbf{Y}_i)$ with $\mathbf{X}_i \in \mathbf{X}$ and $\mathbf{Y}_i = f(\mathbf{X}_i)$.

The data used in by the machine learning algorithm is commonly called *training data*. The output of the learning algorithm, i.e., the function $\tilde{f}$, is called *model*. The model is supposed to represent the task f well, which means that machine learning goes beyond simply remembering the training data. The model is also supposed to make good predictions for $x_i \notin \mathbf{X}$, i.e., for *unseen* input data. The ability to also predict the desired output for unseen input data is called *generalization*. Learning tasks from data is a key aim of machine learning and, hence, the second essential part of machine learning. However, data, as discussed in Chapter 5, is similarly important, since learning from data is not possible without data. We discuss the third essential part, evaluation, in Chapter 7.

Incidental Supervision

Supervised learning works best (i.e., generalizes best) if a lot of training data is available and the desired task is well covered by the samples in the dataset. For many problems, however, collecting a sufficiently large set of $(\mathbf{X}_i, \mathbf{Y}_i)$ pairs is too expensive or not possible for other reasons. We have, for example, already discussed in Chapter 5 that creating reference summaries ($\mathbf{Y}_i$) for sets of input documents (i.e., $\mathbf{X}_i$) is an expensive and difficult task. An insufficient amount of training data leads to poor performance (i.e., a poor approximation of $\tilde{f}$) which limits the applicability of supervised learning in such scenarios.

A solution approach for this problem is to use machine learning to learn signals which help to perform a task from data without supervision, i.e., without providing a large set of input-output pairs. The algorithm learns to perform the task indirectly. Formally, let $\mathfrak{L}$ be a machine learning algorithm which learns from data $\mathbf{D} \in \mathscr{D}$ (which is not necessarily related to the dataset for the task at hand) and let h be a function with $h : \mathfrak{L}(\mathscr{D}) \rightarrow \mathscr{Y}$ learned by $\mathfrak{L}$ on dataset $\mathbf{D}$. The output $\mathbf{Y}_i$ for the task at hand can then be estimated based on the output provided by $\mathfrak{L}$ (which has been trained on data $\mathbf{D}$) by setting $\mathbf{Y}_i = h(\mathbf{X}_i)$.

The problem from Chapter 1 can be considered as a simple example. Suppose the task is to select the most importance information nugget, i.e., $\mathbf{X}_i$ contains the three sentences A, B and C, and $\mathbf{Y}_i$ contains the correct output, i.e., sentence B. Unfortunately, U.S. presidential elections occur only rarely. Hence, we assume that we do not have sufficient training data to train a supervised machine learning model directly (i.e., learning $\tilde{f}$ is not possible).

Suppose that we have access to a dataset $\mathbf{D}$ with social media comments and that we know that events which are discussed heavily in social media tend to be more important. A machine learning model $\mathfrak{L}$ can analyze $\mathbf{D}$ to learn how many comments the events described by sentences A, B and C have in the social media dataset. The knowledge learned by $\mathfrak{L}$ (in this example the number of social media comments) can then be used in function h in Equation 6.1 to select the most important sentence. Formally,

$$h(\mathbf{X}_i) = \underset{\mathbf{s}_j \in \mathbf{X}_i}{\arg\max}\, \mathrm{num}(\mathbf{s}_j) \tag{6.1}$$

where $\mathrm{num}(\mathbf{s}_j)$ denotes the number of times sentence $\mathbf{s}_j$ has been discussed in the social media dataset $\mathbf{D}$.

Note that no supervision about the importance prediction problem is provided to the machine learning model $\mathfrak{L}$. Hence, $\mathfrak{L}$ does not know that it learns something which can be used to help to predict information importance in this case. Assuming that $\tilde{f}$ is provided by a human in this example, only the combination of the knowledge learned by $\mathfrak{L}$ and the knowledge provided by the human can be used jointly to solve the task.

We continue by reviewing prior work for context-free information importance.

6.2 Prior Work for Context-Free Information Importance Estimation

Most summarization models rely on contextual features for importance estimation as discussed in Section 4.2. There are, however, also some works that discuss context-free importance estimation. Usually, they use the idea of context-free information not as the main guiding principle but only as additional features. We review these works in the following.

Manually specified cue words such as „significant“, „impossible“, and „hardly“ have been used to identify important sentences (Edmundson, 1969). Since cue words appear in the sentences directly, this method is context-free in the sense of Definition 22. However, Edmundson (1969) also uses other signals such as sentence location and word overlap with the title, which are not context-free. This fact renders the proposed system to be context-dependent.

TF-IDF scores of words have also been used to identify important words (Brandow et al., 1995; Meade, 1997). Since the term frequency depends on the document at hand, using TF-IDF is no context-free importance feature. However, the IDF score is learned based on a background corpus and might already be helpful to detect important sentences. Sentences that contain rare words might be interesting since they contain words that are usually not contained in articles. Using only the IDF score (without computing the TF score) is a context-free importance estimator.

Hong and Nenkova (2014) find (along with document-dependent features) words that are likely to appear in a summary if they are contained in the source document. To this end, they learn two language models. The first model is trained on a large set of input documents, and the second language model is trained on the set of corresponding output documents. The difference and the ratio between the two resulting word probabilities is computed. Additionally, they calculate the Kullback-Leibler divergence between the word distributions. Since the Kullback-Leibler divergence is not symmetric, they compute the divergence for both combinations of probability distributions. Hence, one divergence is high for words which are likely to occur in summaries whereas the other divergence is high for words which are likely to be not included in the summaries.

PriorSum (Cao, Wei, Li, et al., 2015) extracts context-free features from word vectors contained in sentences with a convolutional neural network. PriorSum uses a regression framework to learn to predict utility scores for sentences. As features, the context-free features are combined with document-dependent features. Hence, PriorSum is not a context-free summarization system.

Cheung and Penn (2013) investigate if domain-specific *caseframes* can be helpful for abstractive summarization. Caseframes are shallow approximations of semantic roles and are derived from the dependency parse of a sentence. They argue, similarly as we argue in this thesis, that current summarization systems have been heavily optimized towards centrality and lexical-semantical reasoning and that in-domain documents can provide helpful knowledge to learn better which information might be important.

X. Huang, Wan, and Xiao (2011) adapt the model presented by D. Wang et al. (2009) to investigate the problem of comparative news summarization which aims at creating summaries containing differences between two related topics. The FIFA World Cups in 2014 and 2018, for example, are related. Hence, the summaries for both events might look similar except for the information which changed from 2006 to 2010. For example, for both events a summary may contain the winner of the tournament. Germany won in 2014 whereas France won 2018. The model uses TF-IDF (TF computed for the source documents) weights to identify important concepts. Even though the approach uses document-dependent features to create summaries, it is an example of how domain-dependent knowledge can be used in summarization.

Kedzie et al. (2015) use a large set of features to estimate sentence importance for disaster summarization. Basic features include sentence length, position, and the number of named entities. They, however, also use domain-specific context-free features. They train a language model on domain-specific Wikipedia pages to learn how domain-specific summary sentences look like (e.g., for earthquakes or shootings). Since they summarize an event over time, they also compute how bursty terms are by computing the change of their TF-IDF scores. Hence, the complete model is context-dependent. A Gaussian process regression model is used to predict sentence salience based on the features.

Gao et al. (2013) select tweets from sets of tweets that have the highest number of re-tweets. Since this strategy is independent of the context (i.e., other tweets in the set of tweets), it can be considered to be a context-free importance estimator.

The summarization system Carpanta (Alonso, Castellón, Casas, & Padró, 2004) summarizes e-mails with a predefined algorithm. The system classifies e-mails according to different kinds of elements present in the mail. For example, the system checks if a mail contains a question, a bullet point list, or if an attachment is described. Based on this classification, different summarization strategies are applied. In the previous examples, the question, the sentence preceding the list or the segment describing the attachment are extracted, respectively.

Sentence length and stopword ratio are two context-free features that also have been used (Cao, Wei, Li, et al., 2015; Christensen et al., 2013; Kedzie et al., 2015). However, these features are similar to the already discussed features usually used in combination with document-dependent features.

We summarize that there are already a few works that make use of context-free features. The context-free features are, however, usually used in combination with context-dependent features. This approach is reasonable if newswire articles have to be summarized which contain strong context-dependent features. To extract the most important information from heterogeneous sources successfully, it is appealing to develop models that are fully context-free, i.e., which do not use any context-dependent features. We present in the next section a fully context-free model.

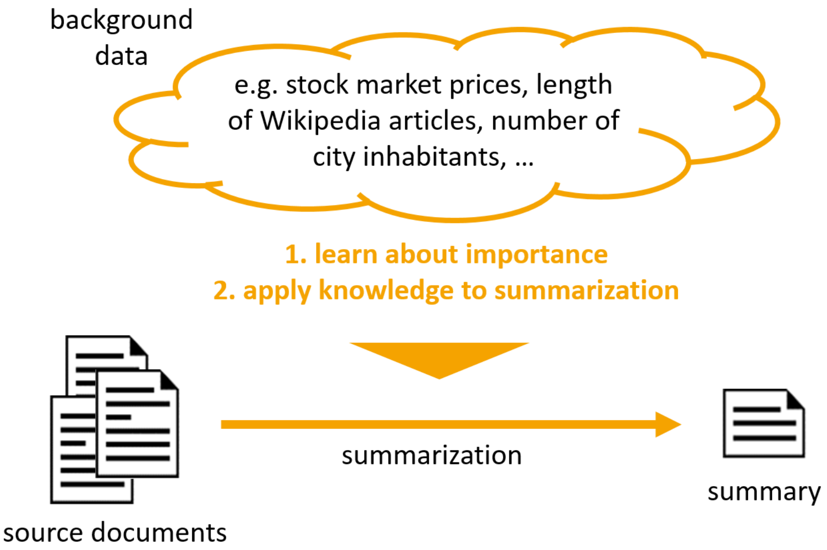

background data
e.g. stock market prices, length of Wikipedia articles, number of city inhabitants, ...
1. learn about importance
2. apply knowledge to summarization
summarization
summary
source documents

Figure 6.2.: Illustration of a context-free information importance estimation model. The model learns in the first step from background data to estimate information importance and applies the learned knowledge in a second to a summarization problem at hand. Since the model learned in the first step to estimate information importance, it does not rely on information importance signals derived from the source documents.

6.3 A Model for Context-free Information Importance Estimation

We present in this section a machine learning approach for importance estimation which is able to estimate information importance for sentences $\mathbf{x}_j \in X_i$ by just analyzing $\mathbf{x}_i$ without analyzing the surrounding sentences $\mathbf{x}_k \in X_i, j \neq k$ or other documents. Hence, it is a context-free information importance estimator according to Definition 22.

To be able to estimate information importance context-free, the model learns to estimate information importance from background data before it is applied to a summarization scenario. The idea is illustrated in Figure 6.2.

The background data does not have to be generated for a particular problem in mind. In the illustration, we list as examples the stock market price of a company, the length of a Wikipedia article for a person, or the number of city inhabitants. The three signals can be learned from data in a first step and can then be potentially used in a second step to estimate information importance. The stock market price might correlate with the importance of news about a particular company, the length of a Wikipedia article might correlate with the importance of a person, and the number of city inhabitants might correlate with the importance of news about a city. Concretely, news about big companies such as Amazon might be interesting to more people than news about a small company, news about Donald Trump might be interesting to more people than news about an only locally known mayor, and news about New York might be interesting to more than news about Darmstadt.

How the knowledge is learned concretely lies, however not in the main focus of this work. Most important is the fact that the learned knowledge can be used in turn to estimate information importance in a context-free manner. Formally, let $(\mathbf{X}, \mathbf{Y})$ be the summarization dataset which has to be summaries (i.e., the test set). To enable the model to summarize the input documents $\mathbf{X}$, we use an independent background dataset $\mathbf{B}$ to train the model. After the model has been trained, it will not have to analyze sentences in $\mathbf{X}_i$ to extract the most importance sentences from $\mathbf{X}_i$ (except of sentence $\mathbf{x}_j \in \mathbf{X}_i$. Hence, the model is able to estimate the importance of $\mathbf{x}_j$ context-free. In this study, we use a large single-document summarization dataset to learn to estimate information importance in a multi-document summarization corpus.

The model estimates the importance of sentences in the test data based on *elements*, which can be extracted from sentences. Different concrete instantiations of elements are possible. Individual words, for example, are rather simple elements that can be extracted from sentences whereas more complex elements are named entities, frame annotations, or discourse markers. We focus in the first version of the model on simple bigrams in Section 6.3.4 and investigate more complex instantiations in Section 6.3.5. The advantage of using bigrams is the fact that they can easily be extracted and have a broad coverage whereas automatically detecting named-entities, for example, is error-prone, and the coverage of detected named-entities might be low.

We model the importance of elements with pairwise preferences (Fürnkranz & Hüllermeier, 2010). A pairwise preference $a \succ b$ simply models that element a is preferred over element b. Preferences may be probabilistic, i.e., the probability that $a \succ b$ rather than $b \succ a$ is $\Pr(a \succ b) \in [0, 1]$, and it holds that $\Pr(a \succ b) + \Pr(b \succ a) = 1$.

Consider the example from Chapter 1 with information nuggets A: ”the U.S. Congress will certify the results on January 6, 2017”, B: ”Donald Trump won the election and will become the 45th president”, and C: ”there are rumors about Russian interferences in the elections”. The journalists prefer to include sentence B over sentence A, which can be modeled with the pairwise preference $B \succ A$. Similarly, $B \succ C$ can be used that the journalist prefers to report information B over information C.

6.3.1 Learning Basis Element and Sentence Importance

Learning Element Importance

As described above, we use a single-document summarization corpus $\mathbf{B}$ as background corpus to train the model. The single-document summarization corpus consists of pairs of input-output documents, i.e., $\mathbf{X}_i$ is an input document and $\mathbf{Y}_i$ is the corresponding summary. Let ζ be a function which extracts all *elements* from a text document. For a given document D, $\zeta(d)$ can, for example, be a set of all bigrams, named-entities, or frames which are contained in document D. As described above, we use bigrams in this study for simplicity.

Let $(\hat{\mathbf{X}}_k, \hat{\mathbf{Y}}_k)$ be a pair of input document and reference summary in the background corpus $\mathbf{B}$. For each element pair (e_i, e_j), for which it holds that element e_i occurs in the summary as well as in the source document (i.e., $e_i \in \zeta(\hat{\mathbf{X}}_k)$ and $e_i \in \zeta(\hat{\mathbf{Y}}_k)$), and element e_j occurs in the input document but not in the summary (i.e., $e_j \in \zeta(\hat{\mathbf{X}}_k)$ and $e_j \notin \zeta(\hat{\mathbf{Y}}_k)$), we generate a preference $e_i \succ e_j$. The intuition of the preference is that e_i has been preferred over e_j to be promoted from the input document into the summary. We add all preferences generated based on the k-th topic in the background corpus to a list of preferences P_k. Based on the per-topic preference lists, we generate a global list of pairwise preferences $P = \bigcup_{k=1}^{|\mathbf{B}|} P_k$ for the whole background corpus where $|\mathbf{B}|$ denotes the number of topics in dataset $\mathbf{B}$.

Based on the collected pairwise preferences P, we estimate the probability that e_i is perfected over e_j with

$$\Pr(e_i \succ e_j) = \frac{\text{num}(e_i \succ e_j)}{\text{num}(e_i \succ e_j) + \text{num}(e_j \succ e_i)} \tag{6.2}$$

where $\text{num}(e_i \succ e_j)$ indicates how many times the preference $e_i \succ e_j$ is contained in the collection of generated preferences P. Based on the probability of e_i to be preferred over another element e_j in Equation 6.2, we define the utility of element e_i to be the average probability over all other elements:

$$u(e_i) = \frac{1}{|e|-1} \cdot \sum_{j=1, j\neq i}^{|e|} \Pr(e_i \succ e_j) \tag{6.3}$$

where $|e|$ denotes the total number of individual elements.

Inferring Sentence Importance

We estimate the utilities of sentences by averaging the utilities of elements which are contained in sentences $\mathbf{s}_i \in \mathbf{X}_k$ (stemming from the summarization topic $(\mathbf{X}_k, \mathbf{Y}_k)$). Formally, we set

$$v(\mathbf{s}_i) = \frac{1}{|\zeta(\mathbf{s}_i)|} \cdot \sum_{e \in \zeta(\mathbf{s}_i)} u(e) \tag{6.4}$$

where $|\zeta(\mathbf{s}_i)|$ indicates the number of elements contained sentence $\zeta(\mathbf{s}_i)$. Note that the definition of $v(\mathbf{s}_i)$ does not depend on $\mathbf{X}_k$ and is therefore a context-free importance estimator according to Definition 22. Removing or adding sentences to the input documents or randomizing the order of the sentences, for example, does not change the utility of sentences.

6.3.2 Contextual Pairwise Preferences

The proposed utility function v can be used with a greedy sentence selection algorithm with redundancy avoidance such as listed in Algorithm 4 and Algorithm 5 in Section 3.3. It can, however, not be used as a utility function to rank sentences for a greedy sentence selection without redundancy avoidance (see Algorithm 3) since the utility function does not consider redundancy. Two sentences that are considered to be important and are therefore ranked highly will both be selected even if they share a lot of content. An alternative solution for using a redundancy-aware sentence selection algorithm is to incorporate redundancy already in the estimated utilities. To this end, we propose *contextual preferences* which are able to model importance and redundancy (with respect to a partially generated summary) jointly. Instead of learning $e_i \succ e_j$, which means that element e_i is preferred over element e_j, we propose the notation $e_i \succ e_j \mid C_k$ to denote that $e_i \succ e_j$ holds in a particular context C_k but might not hold in another context C_l.

Consider again the example from Chapter 1 with a journalist preferring B over A, C over A, and B over C. If there is only space left for one information nugget, the journalist includes nugget B in the article. If, however, two nuggets can be included, it would not be reasonable to include nugget B twice. If the journalist includes B in a first step, he or she might include C in a second step to not include redundant information. This observation means that nugget C might be preferred over B, given that B has already been included in the news article. Formally, B is selected in the first step because of $B \succ C$. In the second step, C might be preferred due to $C \succ B \mid B$.

To define the number of observations $e_i \succ e_j$ for a context C we slightly modify the generation of the preference collection P. Instead of iterating over all topics in $\mathbf{B}$, we restrict the search for preferences to topics $(\hat{\mathbf{X}}_k, \hat{\mathbf{Y}}_k)$ in which the summary $\hat{\mathbf{Y}}_k$ contains the context C. We then remove the elements in context C from $\hat{\mathbf{Y}}_k$ and continue with the computation of $\Pr(e_i \succ e_j)$, u, and v as described in Equation 6.2 and Equation 6.4. The model learns in contextual situations which elements should be added next given that some elements are already contained in the partial summary. To mitigate sparsity issues, we consider the elements in C individually. It would otherwise be difficult to find summaries which contain all the elements in C.

Note that even though contextual pairwise preferences depend on a context, the resulting utility function is still context-free in the sense of Definition 22 since the importance of a sentence is still independent of other content contained in the input document(s) and is just adapted to the information which is already contained in the partially created summary.

6.3.3 Example

The following example illustrates the behavior of the redundancy-aware pairwise preferences. Let the background dataset $\mathbf{B}$ contain only two topics. Let the first topic's input document be $\hat{\mathbf{X}}_1 = (a, b, c)$ and the corresponding summary be $\hat{\mathbf{Y}}_1 = (a, b)$. Let furthermore the second topic consist of input document $\hat{\mathbf{X}}_2 = (a, b, c)$ and summary $\hat{\mathbf{Y}}_2 = (a)$. Since a and c occur in the input document but only a also occurs in the summary, the model generates the pairwise preference $a \succ c$. Similarly, the model generates the preference $b \succ c$. Hence, the set of preference for the first topic equals $P_1 = (a \succ c, b \succ c)$. $P_2 = (a \succ b, a \succ c)$ are generated based on the second topic. The resulting probabilities and utilities are listed in Table 6.1. Given that a summarization topic in $\mathbf{D}$ contains the three elements a,b, and c, a will be selected in the first iteration of a greedy selection process. In the second iteration, however, a is not selected again since different preferences are generated given that a has already been selected in the first iteration (i.e., given context a).

First iteration (with empty context):

topic	input $\hat{\mathbf{X}}_k$	output $\hat{\mathbf{Y}}_k$	context C	preferences	probabilities	utilities
k=1	a, b, c	a, b	$\emptyset$	a $\succ$ c, b $\succ$ c		
k=2	a, b, c	a	$\emptyset$	a $\succ$ b, a $\succ$ c		
					$\Pr(a \succ b) = 1$ $\Pr(a \succ c) = 1$ $\Pr(b \succ c) = 1$	$u(a) = 1$ $u(b) = 0.5$ $u(c) = 0$
					→ a is selected	

Second iteration (after a has already been selected):

topic	input $\hat{\mathbf{X}}_k$	output $\hat{\mathbf{Y}}_k$	context C	preferences	probabilities	utilities
k=1	a, b, c	a, b	a	b $\succ$ a | a, c $\succ$ a | a		
k=2	a, b, c	a	a	$\emptyset$		
					$\Pr(a \succ b) = 0$ $\Pr(a \succ c) = 0.5$ $\Pr(b \succ c) = 1$	$u(a) = 0.25$ $u(b) = 1$ $u(c) = 0.25$
					→ b is selected given context a	

Table 6.1.: Example of contextual pairwise preferences.

6.3.4 Information Importance Estimation with Bigrams

Data

The assumption of many summarization systems is that information frequency, and information location can be used to estimate information importance. While this may be a reasonable assumption for some document collections (such as newswire documents), we suspect that the assumption may not hold for heterogeneous document collections. We modify the DUC 2004 multi-document summarization corpus (see Section 5.2 for details) to simulate a messy heterogeneous dataset by *shuffling* and *oversampling* to remove commonly used indicators for importance from the dataset. We do this to demonstrate that many document summarization algorithms fail, whereas the proposed context-free model will maintain its performance.

In order to remove the location signal, we randomly shuffle the sentences to hide the very strong sentence position signal, which is commonly used to detect importance in news documents. With oversampling, we aim at hiding important information in the corpus by increasing the frequency of unimportant information. To perform the oversampling for topic $\mathbf{X}_i$, we search for sentence $\hat{\mathbf{s}}$ with

$$\hat{\mathbf{s}} = \arg\min_{\mathbf{s}\in\mathbf{X}_i} \sum_{\mathbf{s}_i\in\mathbf{X}_i} \mathrm{sim}(\mathbf{s}, \mathbf{s}_i), \tag{6.5}$$

Dataset	avg. sim
DUC 2004	0.088
DUC 2004	0.088
DUC 2004 200%	0.069
DUC 2004 500%	0.062
DUC 2004 1000%	0.060

Table 6.2.: Average similarities of the sentences contained in the test corpora.

where sim is a similarity measure for two sentences, and add $\hat{\mathbf{s}}_i$ to a random document in topic $\mathbf{X}_i$. We perform this operation multiple times. Since we duplicate the sentences, we make sure that we do not introduce new, important information to the corpus which is not contained in the summary. We use the similarity measure in Equation 6.6 in our experiments, where cos is a cosine similarity implemented in the DKPro Similarity framework (Bär, Zesch, & Gurevych, 2013) with TF-IDF values based on English Wikipedia articles, and jacc denotes to the well-known Jaccard measure. This simple combination lead to reasonably good results on the English subtask of the SemEval 2014 Semantic Textual Similarity dataset.[1]

$$\mathrm{sim}(s_i, s_j) = \frac{\cos\ \mathrm{sim}(s_i, s_j) + \mathrm{jacc}\ \mathrm{sim}(s_i, s_j)}{2} \tag{6.6}$$

With this methodology, we create four new corpora with 100%, 200%, 500%, and 1000% of the size of DUC 2004. In the 100% corpus sentences are only shuffled without duplicating sentences. With increasing corpus size, prior frequent information can not be detected easily anymore. An analysis of the result of the oversampling is given in Table 6.2. The average similarity decreases with increasing size, which confirms that oversampling successfully creates a dataset with more equally frequent sentences.

Reference Systems

We compare the new algorithm, *CPSum*, with two baselines and two well-known summarization algorithms.

The first baseline *Optimal* has access to the reference summaries. Hence, it is no fair competitor for the remaining systems. Nevertheless, it provides useful information about the maximal reachable score for each dataset. Since computing the exact optimal score is computationally expensive, we only provide a pseudo-optimal value computed by applying a greedy search.

The first fair baseline system, *Random*, selects sentences from the source documents randomly. It does not have access to the reference systems. Hence, it is the first system that can be compared with the previously discussed systems. Since the most important information in newswire documents is often contained in the first sentences, just selecting the first few sentences as a summary is a strong baseline.

[1] http://alt.qcri.org/semeval2014/

Hence, we use *Lead* as a second baseline to provide evaluation scores for a system, which selects the first sentences of each document.

We use *Centroid* (Radev, Jing, & Budzikowska, 2000) as a representative system for centroid-based systems. To generate the summaries for this approach we apply the widely used MEAD system (Radev et al., 2004), in which *Centroid* is implemented. For *Centroid* we use the default linear feature combination, length cutoff and re-ranker. As a competitive state-of-the-art representative for graph-based approaches (Hong et al., 2014) we apply *LexRank* (Erkan & Radev, 2004), which is also implemented in the MEAD system. For *LexRank* we used the LexRank feature, standard length cutoff and the default re-ranker.

Using CNN/DailyMail as Background Corpus

Since *CPSum* learns about the importance of objects from a background corpus, we need a concrete instantiation for the abstract objects and second a background corpus to learn from. As instances for the objects for which we learn contextual preferences of the form $a \succ b \mid C$ we use bigrams of stemmed words, which means that *CPSum* will learn about the importance of bigrams. Hence, the context C is a sequence of bigrams. As mentioned above, any other linguistic entity like named-entities, opinions, or events would also be possible choices.[2] We use bigrams of stemmed words since they do not need any sophisticated preprocessing.

To learn the importance of bigrams, we use the CNN/DailyMail corpus (Hermann et al., 2015), a background corpus initially created for question answering. Although this corpus does not provide well-written summaries for each article but only sentence-length bullet points summarizing the content of the article, we can use this information to derive the necessary training signals for learning object importance. The corpus contains about 100k CNN document-summary pairs crawled from CNN and about 197k pairs crawled from DailyMail. For training, we use a subset of 100k randomly selected documents.

Since most of the bigrams contained in the background corpus do not appear in a given input document set, we apply a lazy learning strategy to only learn to estimate the importance of elements that appear in the test data. Furthermore, we only learn preferences for contexts that we actually observe during summarization. This approach decreases the learning effort significantly. Note that we use only information which is available at test time and do not provide any additional information which is not available at test time.

Results

Table 6.3 shows the ROUGE (C.-Y. Lin, 2004) evaluation scores of the tested systems.

First, we observe that the evaluation scores for both, ROUGE$_1$ and ROUGE$_2$ recall stays nearly constant for the oracle system *Optimal*. From this result, we conclude that our modifications did neither remove from nor add important information to the corpus. After the modifications, it is still possible to generate

[2] We investigate other choices in Section 6.3.5

	ROUGE$_1$ Recall					ROUGE$_2$ Recall				
	D04	D04-1	D04-2	D04-5	D04-10	D04	D04-1	D04-2	D04-5	D04-10
Optimal	0.4043	0.4043	0.4046	0.4043	0.4044	0.0940	0.0941	0.0943	0.0940	0.0942
Random	0.2955	0.3095	0.2863	0.2736	0.2633	0.0435	0.0463	0.0360	0.0313	0.0322
Lead	0.3424	0.3138	0.2786	0.2636	0.2548	0.0766	0.0524	0.0382	0.0313	0.0282
Centroid	**0.3542**	0.3158	0.3082	0.2690	0.2474	**0.0867**	0.0605	0.0576	0.0396	0.0331
LexRank	0.3231	0.3219	0.3186	0.3052	0.2990	0.0659	**0.0645**	**0.0631**	0.0542	0.0522
CPSum	0.3267	**0.3247**	**0.3264**	**0.3264**	**0.3264**	0.0603	0.0604	0.0617	**0.0617**	**0.0617**

Table 6.3.: ROUGE$_1$ and ROUGE$_2$ scores on the original and the modified DUC 2004 corpora. D04 is shorthand for DUC 2004, D04-1 refers to the dataset with shuffled sentences, and D04-2,5,10 to the datasets with two, five, and ten times the size of the original DUC 2004 dataset.

summaries with a ROUGE$_1$ value of at least 0.40 and a ROUGE$_2$ value of at least 0.09. The performance of *Random* decreases if the corpus size increases since it becomes more likely to pick a poor sentence. The baseline *Lead*, which simply uses the first sentences of each article, is considered to be a strong baseline in newswire documents and is able to summarize the original DUC 2004 data reasonably well. In the modified corpora with a randomized sentence order, its performance is as expected not better than *Random*.

The two state-of-the-art reference systems work well on the original DUC 2004 corpus, where *Centroid* achieves the best results. This behavior is also expected since it uses positional and centrality features, which provide very good signals for importance in the corpus. When these signals are more and more removed in D04-1 – D04-10, we observe a big performance decrease in both ROUGE$_1$ and ROUGE$_2$. *LexRank* behaves similarly to *Centroid* but with a less fast decrease of performance.

CPSum performs moderately at the original DUC 2004 dataset since it does not use the strong importance signals sentence position and sentence centrality. The strength of *CPSum* can be observed in the modified corpora, where the performance stays comparable to the performance at the original corpus and does not decrease as it can be observed for the other approaches. In terms of ROUGE$_1$ scores, *CPSum* has the best performance on the four modified corpora. In terms of ROUGE$_2$, its original performance is similar to the performance of *LexRank* but lower than *Centroid*. The performance of *Centroid* drops significantly after shuffling the sentences. If we add more and more sentences with oversampling, the performance of *Centroid* drops again faster than the performance of *LexRank*. *CPSum* outperforms all systems when we increase the amount of noise in the corpora D04-5 and D04-10.

We show an example of the sentences scoring generated with contextual pairwise preferences in Figure 6.3. We display the same sentence twice. At the top, we display the context-free pairwise preference scores of the elements of the sentence by using a darker font for more important information. At the bottom, we show the contextual pairwise preferences scores of the same sentence after adding this particular sentence to the summary. We observe that the importance scores of important elements such as *Osama bin Laden* are adequately estimated. After adding the sentence to the summary, we can see how *CPSum* discounts the scores for different elements differently.

Saudi exile Osama bin Laden , the alleged mastermind of a conspiracy to attack U.S. targets around the world, and Muhammad *Atef,* the alleged military commander of bin Laden's terrorist organization, Al-Qaeda, were charged in a separate *238-page* indictment with murder and conspiracy in the bombings.

Saudi exile Osama bin Laden , the alleged mastermind of a conspiracy to attack U.S. targets around the world, *and* Muhammad *Atef,* the alleged military commander of bin Laden's terrorist organization, Al-Qaeda, were charged in a separate *238-page* indictment with murder and conspiracy in the bombings.

Figure 6.3.: Example of the importance of the elements in a sentence before (top) and after (bottom) adding the sentence to the summary. The darker the font color the more important the element. Elements with less than 100 gathered preferences are displayed in italics.

6.3.5 Information Importance Estimation Capabilities with More Complex Elements

In a second series of experiments, we investigate whether low-level linguistic annotations can improve the performance of the previously proposed model. To this end, we investigate several annotation types that can be used instead of bigrams. We study which types of linguistic annotations prove useful to capture the notion of importance. The underlying hypothesis is that for estimating information importance, linguistic elements that involve abstractions over surface forms should be more apt to generalize to unseen data than surface-oriented features such as bigrams, especially when training resources are scarce or when moving to novel domains. On the other hand, linguistic annotations could also be noisy, sparse, or suffer from being too fine-grained. We explore these questions by injecting knowledge from different element instantiations, such as conceptual frames, the expression of sentiment and opinion, or discourse relations that could reflect importance.

Investigated Annotations

In this section, we describe which types of annotations we investigate in the following experiments, why we suspect that they could be helpful, and if/where downstream applicability has already been investigated. The annotation types are roughly ordered according to increasing complexity.

Unigrams. The unigram annotation indicates if a given word type is present in the text. Unigrams were also used by Hong et al. (2014) for importance estimation.

Bigrams. Bigrams capture consecutive bigrams appearing in the text and have been used before for estimating sentence importance in summarization systems (Carbonell & Goldstein, 1998; Gillick et al., 2009; H. Lin & Bilmes, 2011; Zopf, Loza Mencía, & Fürnkranz, 2016a).

Trigrams. Trigrams are analogous to bigrams but indicate the appearance of a consecutive sequence of three words. Hence, they are able to capture longer phrases.

Verb Stems. For each verb in the text we use its lemma as a feature (e.g., *killing, killed* → *kill*). The intuition here is that particular verbs convey importance better than others. If a news article contains the information that someone has been killed, this information will most likely also be contained in the reference summary. On the other hand, it is often reported in news articles that person x said y (e.g., uttered an opinion) which might be a rather unimportant detail which is not contained in the summary.

Chunks. A recent study on interactive summarization (P. V. S. & Meyer, 2017) shows that chunks can also be used as an alternative to bigrams in a summarization system. Chunks are constituent parts of a sentence with a specific grammatical meaning (e.g., noun chunks, verb chunks). In this work we use the Tree-tagger chunker[3] and consider four chunk types, namely noun chunks (*NC*), verb chunks (*VC*), adverbial chunks (*ADVC*) and adjectival chunks (*ADJC*). As chunks capture grammatical meaning, we suspect that they are a viable replacement to bigrams and can capture richer importance features.

Named entities (NEs). This annotation type identifies mentions of entities and their semantic types, such as persons, locations, or organizations. We evaluate different variants of NE annotations. For example, the 21 entity types found by applying the CoreNLP named entity recognizer (Manning et al., 2014), e.g., *PERSON*, *CITY*, or *COUNTRY*; 91 fine-grained entity types from the FIGER type inventory (Ling & Weld, 2012), e.g., */person/politician* or */building/hotel*; and unique IDs for each entity. We obtain Freebase (Bollacker, Evans, Paritosh, Sturge, & Taylor, 2008) entity IDs via an entity linking system (Heinzerling, Judea, & Strube, 2015) and then map these IDs to their FIGER type.

Frames. Following the theory of frame semantics (Fillmore, 1976), humans understand the meaning of words in terms of frames. FrameNet (Baker, Fillmore, & Lowe, 1998) provides an inventory of such frames that are used to provide a fine-grained interpretation of predicates in sentences by disambiguating the predicate's meaning with respect to frames. FrameNet annotations have been used for question answering and automatic text summarization (Gildea & Jurafsky, 2002), event detection (Spiliopoulou, Hovy, & Mitamura, 2017), text understanding (Fillmore & Baker, 2001), and textual entailment (Aharon, Szpektor, & Dagan, 2010). To annotate all nouns and verbs of the texts with frames we use the neural network-based system presented by Hartmann, Kuznetsov, Martin, and Gurevych (2017) which assigns a frame to a word based on the word itself and the surrounding context in the sentence.

Concepts. In order to identify concepts in text documents, we follow the work of Falke, Meyer, and Gurevych (2017) which primarily relies on open information extraction (Mausam, Schmitz, Bart, Soderland, & Etzioni, 2012) to detect mentions of concepts and their relationships, and then use several measures of semantic similarity between them to cluster mentions of the same concept together. In this work, we use small concept clusters obtained by string matching different mentions (concepts string) and broader clusters based on semantic similarities (concepts sim). Compared to bigrams, which are directly defined on the lexical level, we expect the clustering to semantic groups to yield richer importance signals that generalize better.

Connotation frames (CFs). Connotation frames are a new formalism for analyzing subjective roles and relationships implied by a given predicate (Rashkin, Singh, & Choi, 2016). For example, in $\textit{[Brazil]}_{agent}$ *is suffering from* $\textit{[a failing economy]}_{theme}$, the verb *suffering* indicates that the writer treats *Brazil* more sympathetically and the theme more as an "antagonist". *Brazil* most likely feels negatively towards

[3] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

the theme, it has been hurt, its "mental health" is distressed, but it is considered valuable. All these relationships are unified by a connotation frame that contains labels for relationships inferable from the predicate *suffer*. Given that connotation frames capture the implicit sentiment of the writer and sentiment between entities, we suspect that connotation frames can help to signal importance.

Discourse Relation (DR) Senses. Discourse Relations as annotated in PDTB (Prasad et al., 2008) indicate specific thematic relations between clauses or sentences in discourse, such as *causation, contrast, or concession*. These relation *senses* can be explicitly marked (e.g., *but, whereas*) or are implicitly understood in unmarked juxtaposition of sentences. DRs capture a notion of *importance* at the level of text organization that is especially relevant for summarization. We use annotations from the output of the DR sense classification system presented by Mihaylov and Frank (2016).

Sentiment. To investigate whether sentiment words can signal importance, we use all sentiment words retrieved from the sentiment lexicon presented by M. Hu and Liu (2004).

Experimental Setup

For the experiment, we use three well-known multi-document summarization datasets, namely the DUC 2004 (DUC2004), TAC 2008 (TAC2008), and TAC 2009 (TAC2009) corpora (see Chapter 5 for details).

We estimate sentence scores based on the text element utilities defined in Section 6.3.1 depending on whether ROUGE recall or ROUGE precision scores have to be predicted. ROUGE is a simple similarity measure that estimates the similarity of two texts based on the overlap of n-grams (C.-Y. Lin, 2004). The more n-gram which are contained in a reference summary are contained in the sentence, the higher the ROUGE recall is. The higher the number of n-gram in a sentence is which also appear in the reference summary, the higher the ROUGE precision score is.[4] For ROUGE recall, we simply add the utility scores of all text elements that appear in a sentence **s** to compute the sentence utility score $\sum_{e \in \zeta(\mathbf{s})} u(e)$. For ROUGE precision, we divide the score for ROUGE recall by the length of the sentence: $\frac{1}{|\mathbf{s}|} \sum_{e \in \zeta(\mathbf{s})} u(e)$.

Ranking Experiments

We now provide the results of three different ranking experiments according to the evaluation strategy proposed in Section 3.5.2. To generate target rankings, we extract all sentences from the input documents and rank the sentences according to their ROUGE recall and ROUGE precision score.[5] Results for ROUGE precision and ROUGE recall are in columns P and R, respectively. We report Kendall's Tau, nDCRS (see Section 3.5.2 for details), and precision@k (with $k = 20$ in all experiments) scores for all annotation types including bigrams which were described in the previous section. For better visualization, we highlight the best five results in every column. For the first two experiments, higher scores are

[4] ROUGE, precision, and recall are discussed in more detail in Chapter 7.

[5] We discuss why the usually ignored ROUGE precision score is more reasonable regressand than ROUGE recall in Section 6.4 if a greedy selection is used and ROUGE recall scores are more reasonable regressands if ILP-based methods are used. Hence, the ranking performance according to both scores is interesting.

	Kendall’s Tau		nDCRS		precision@k	
	P	R	P	R	P	R
bigram	**.504**	**.634**	**.536**	**.929**	**.536**	**.588**
cf-effect-object	-.049	.266	.085	.686	.085	.229
cf-state-subject	-.044	.292	.088	.703	.088	.240
chunk-concepts	**.343**	**.478**	**.363**	**.861**	**.363**	**.444**
concepts-string	**.181**	.276	**.259**	.699	**.259**	.263
concepts-sim	.130	.270	.168	.698	.168	.247
connotation-frames	-.006	.326	.091	.734	.091	.266
entity-importance	-.065	-.027	.110	.532	.110	.194
entity-links	.187	.320	.232	.752	.232	.329
entity-type-coarse	.037	.087	.136	.577	.136	.157
entity-type-corenlp	.091	.352	.152	.758	.152	.315
entity-type-figer	.130	.275	.179	.714	.179	.248
entity-type-fine	.129	.274	.181	.713	.181	.252
FN-frames	.052	.399	.118	.785	.118	.317
FN-frames-nounsOnly	.154	**.487**	.168	**.848**	.168	**.402**
FN-frames-verbsOnly	.016	.216	.099	.646	.099	.186
sentiment-annos	.097	.230	.209	.693	.209	.263
discours-rel	.009	.232	.134	.644	.134	.173
trigram	**.314**	**.535**	**.443**	**.866**	**.443**	**.426**
unigram	**.401**	**.693**	**.350**	**.931**	**.350**	**.568**
verb-stem	.088	.275	.147	.702	.147	.251

Table 6.4.: Results of the ranking experiments evaluated on training data

better. For the ablation experiments, lower scores are better. Details and analysis of the experiments are provided in the next three paragraphs.

Which Annotation Types Can Potentially Convey Importance?

In the first experiment, we investigate whether the annotation types can potentially be used by the model to learn about the importance of information. To this end, we use in this experiment the same data for training and testing. If the model is able to learn based on the annotations, it should be able to achieve reasonable performance in this setting. We provide the results of this experiment in Table 6.4.

The best performing annotation types are unigrams, bigrams, trigrams, and chunk-concepts. This result is not surprising in this experimental setup since we test the performance on the training data. This result means that annotations that are close to the text are able to adapt well to the data. Similarly, chunk-concepts and concepts-string perform reasonably well. Very surprising is the large performance difference of FN-frames-nounsOnly between ROUGE precision and ROUGE recall ranking prediction (columns P and R). FN-frames-nounsOnly annotations do not perform well for ROUGE precision but are among the best annotations for ROUGE recall. We observe that cf-effect-object and cf-state-subject perform worst, which can be explained by the fact that these annotations occur very frequently in both source documents and summaries. Hence, the model is not able to use these annotations as importance indicators. We annotated all text elements automatically, meaning that higher-level annotations might

	Kendall's Tau		nDCRS		precision@*k*	
	P	R	P	R	P	R
bigram	**.306**	**.539**	**.253**	**.863**	**.253**	**.424**
cf-effect-object	-.051	.269	.083	.687	.083	.230
cf-state-subject	-.054	.284	.083	.697	.083	.234
chunk-concepts	**.175**	**.367**	**.206**	**.773**	**.206**	**.298**
concepts-string	.106	.193	.146	.639	.146	.179
concepts-sim	.093	.225	.135	.669	.135	.225
connotation-frames	-.011	.335	.089	.739	.089	.267
entity-importance	-.076	-.060	.107	.510	.107	.193
entity-links	**.135**	.264	**.169**	.709	**.169**	.261
entity-type-coarse	.031	.100	.138	.582	.138	.155
entity-type-corenlp	.075	.358	.132	.766	.132	**.316**
entity-type-figer	.122	.272	.165	.709	.165	.243
entity-type-fine	.117	.269	.163	.708	.163	.236
FN-frames	.027	**.383**	.107	**.772**	.107	.297
FN-frames-nounsOnly	.116	**.474**	.133	**.836**	.133	**.364**
FN-frames-verbsOnly	.010	.209	.096	.639	.096	.186
sentiment-annos	.068	.215	.148	.673	.148	.222
discours-rel	.011	.234	.133	.646	.133	.174
trigram	**.172**	.366	**.186**	.760	**.186**	.241
unigram	**.300**	**.654**	**.260**	**.913**	**.260**	**.515**
verb-stem	.042	.250	.114	.671	.114	.215

Table 6.5.: Results of the ranking experiments on unseen test data

be inaccurate to some extent. This effect might be another explanation for the superior performance of low-level annotations.

Which Annotation Types Convey Importance Across Topics?

In the next experiment, we analyze how well the model is able to transfer learned knowledge to other, unseen topics. From the four datasets used, we perform four experiments in which we select one dataset as the test set and use the remaining three datasets for training. We report the average performance of the four experiments in Table 6.5.

The best three performing annotations are unigrams, bigrams, and chunk-concepts. Compared to the first experiments, we observe that trigrams lost performance. The reason is that it is likely that there are more, unseen trigrams in the test data for which the model was not able to learn importance scores. The ability to generalize of trigrams is limited. Somewhat surprising is that chunk-concepts, which are also very close to the surface text, perform still quite well. Chunk-concepts seem to generalize better than trigrams. FN-frames-nounsOnly did not lose much performance compared to the first experiment, which indicates that FN-frames-nounsOnly does not overfit to the training data and generalizes well. Entity-links, entity-type-corenlp, entity-type-fine, and FN-frames are also among the top 5 in some columns. We also see a rather large relative performance drop of bigrams compared to unigrams, for example.

	Kendall's Tau		nDCRS		precision@*k*	
	P	R	P	R	P	R
bigram	**-.112**	**-.064**	**.124**	**.453**	**.124**	**.093**
cf-effect-object	-.092	-.016	.138	.487	.138	.115
cf-state-subject	-.092	-.016	**.137**	.487	**.137**	.116
chunk-concepts	**-.095**	**-.019**	.139	**.486**	.139	.115
concepts-string	-.087	-.012	.138	.490	.138	.115
concepts-sim	-.089	-.013	**.135**	.490	**.135**	**.114**
connotation-frames	-.092	-.017	.142	.487	.142	**.113**
entity-importance	-.089	-.007	.139	.495	.139	.119
entity-links	-.078	-.005	.142	.496	.142	.116
entity-type-coarse	-.087	-.004	.142	.497	.142	.117
entity-type-corenlp	**-.095**	**-.018**	**.136**	**.486**	**.136**	**.113**
entity-type-figer	-.083	-.007	.142	.494	.142	.115
entity-type-fine	-.083	-.007	.139	.494	.139	.115
FN-frames	**-.095**	**-.018**	.138	**.486**	.138	.115
FN-frames-nounsOnly	**-.095**	**-.018**	.139	**.485**	.139	.115
FN-frames-verbsOnly	-.093	-.017	.138	**.486**	.138	**.114**
sentiment-annos	-.084	-.010	.144	.492	.144	.117
discours-rel	-.092	-.016	.139	.488	.139	.117
trigram	**-.095**	-.017	**.137**	**.486**	**.137**	**.114**
unigram	-.093	-.017	.141	.487	.141	.116
verb-stem	-.093	-.017	.138	**.486**	.138	.115

Table 6.6.: Ablation experiments on unseen test data

Ablation Experiments

In the last ranking experiment, we perform an ablation study within the previous experimental setting, i.e., testing performance on unseen test data. We aggregate the rankings of all annotation types except for one. The first line in Table 6.6, for example, contains the aggregated ranking of all annotation elements except bigrams. The scores indicate how much performance is lost if a particular annotation element is removed from the ensemble. Lower scores are therefore better. As aggregation function, we compute the average rank for each sentence and rank the sentences according to the averaged ranks.

The biggest drop in performance is observed when bigrams or entity-type-corenlp are removed from the ensemble. Entity-type-corenlp contributes to the ensemble even though they did not perform well in the first two experiments. We observe that unigrams do not contribute much to the ensemble even though they showed a very good performance in the first two experiments. Similarly to the second experiment, frame-based annotations show reasonably good performance. Annotations based on connotation frames also rank among the top 5.

Predicting Preference Labels for Source And Reference Sentences

In the next experiment, we evaluate the performance of different annotation types based on the new pairwise prediction evaluation method described in Section 3.5.3. With this evaluation method, we test

	DUC 2003	DUC 2004	TAC 2008	TAC 2009	average
bigram	**0.573**	0.538	0.415	0.445	0.493
cf-effect-object	0.538	0.520	**0.663**	**0.743**	**0.616**
cf-state-subject	0.548	0.439	0.420	0.512	0.480
chunk-concepts	**0.641**	**0.613**	**0.556**	0.602	**0.603**
concepts-string	0.513	0.429	0.371	0.382	0.424
concepts-sim	0.520	0.468	0.438	0.473	0.475
connotation-frames	0.551	0.556	0.546	0.592	0.561
entity-importance	**0.597**	**0.634**	**0.655**	**0.658**	**0.636**
entity-links	0.510	0.450	0.370	0.364	0.424
entity-type-coarse	0.512	0.487	**0.664**	**0.695**	**0.590**
entity-type-corenlp	**0.582**	**0.608**	0.551	**0.616**	0.589
entity-type-figer	0.495	0.487	0.453	0.408	0.461
entity-type-fine	0.497	0.490	0.456	0.405	0.462
FN-frames	0.474	0.497	0.515	0.496	0.496
FN-frames-nounsOnly	0.521	0.537	0.531	0.539	0.532
FN-frames-verbsOnly	0.490	0.487	0.468	0.507	0.488
sentiment-annos	0.430	0.402	0.353	0.356	0.385
discours-rel	0.550	**0.608**	**0.628**	**0.604**	**0.598**
trigram	0.373	0.285	0.210	0.254	0.281
unigram	**0.617**	**0.601**	0.530	0.553	0.575
verb-stem	0.497	0.517	0.515	0.500	0.507

Table 6.7.: Results of the pairwise preference experiments.

how well a model can distinguish between sentences sampled from source documents and summaries. The results of the experiment are displayed in Table 6.7. We use 3 of 4 datasets for training and test on the remaining dataset. We average the results of the four resulting test setups.

The results are very different compared to the previously conducted ranking results. The model is best able to use entity-importance to distinguish between source and summary sentences, followed by cf-effect-object, chunk-concepts, and now also discourse-rel performs consistently well. Bigrams, which performed very well in the ranking experiments, perform poorly in this experiment. They show a bad performance in particular in the TAC 2008 and TAC 2009 corpora. The annotation types cf-effect-object and entity-type-coarse perform well in the TAC datasets. Entity-importance achieves the best overall performance.

6.3.6 Conclusions

In this section, we presented a context-free information importance estimator. CPSum learns from large single-document summarization corpora such as the CNN/DailyMail corpus to perform multi-document summarization, which is only possible because the model does not rely on context-dependent features. Context-free information importance estimation is the key to leveraging large training datasets for summarization tasks where not much training data is available. The training setup which has originally been presented by Zopf, Loza Mencía, and Fürnkranz (2016a) has been adopted since then by many abstractive summarization systems for the so-called *out-of-domain testing*. Note, however, that this is not out-of-domain testing for information importance estimation in general but rather another way to test information importance estimation abilities.

CPSum is able to cope with summarization scenarios where neither information frequency nor structural features such as information location can be used to detect important information. We showed that the performance of conventional text summarization systems decreases in such a setting. Previous approaches can be confused easily by adding more and more irrelevant information, whereas the performance of CPSum stays constant.

The way CPSum learns prior knowledge it is able to detect important information similar to the way human experts address a summarization task since humans also do not rely on frequency or location features but bring along a lot of world knowledge which is used to detect important information.

CPSum is also different in the way it copes with redundancy. Instead of measuring the similarity to already selected sentences such as the majority of the previous systems, it can estimate the utility of elements with contextual preferences. This feature enables CPSum not only to detect redundancy but also to use synergy effects between sentences. Hence, adding one sentence to the summary can also increase the utility of other sentences. Furthermore, our system can be adapted to different user interests by learning from other source documents.

We studied in the second series of experiments if and how well a wide range of linguistic annotations can improve the performance of the previously proposed summarization model.

The ranking experiments show that annotations that are close to the surface text such as *n*-grams and chunks perform best. Simple annotations can also serve as simple annotations to build strong baselines. However, other annotations also showed potential in specific situations. In particular, entity and frame annotations are able to improve the performance in some cases.

In our pairwise preference prediction experiments, we observed a different behavior. Bigrams, which performed well for ranking, did not perform well in this experiment, which is surprising. Instead, entity-based annotations, connotation information, and discourse relations perform well in distinguishing source sentences from reference sentences.

6.4 Sentence Regression

The model presented in Section 6.3 learns from a background corpus signals which are used in Equation 6.3 and Equation 6.4 to estimate the importance of information. In this model, Equation 6.3 and Equation 6.4 are hand-crafted and not learned. The model never receives any feedback how well it performs. More specifically, the model does not know the problem to be learned during training and is therefore not able to improve its performance in a target-oriented manner during the training phase.

In the following section, we present a learning setup called sentence regression in which a machine learning model receives feedback on how well it performs. The idea of sentence regression, which is a special kind of supervised learning where $Y \subseteq \mathbb{R}$, is to learn to estimate utility scores of sentences. Based on the feedback, the model can adapt its behavior and improve its performance for the task.

As discussed previously in Section 3.3.2, the ranking $\mathbf{R}$, which is usually generated with the help of a sentence utility function u, is crucial for the performance of a greedy sentence selection strategy. Hence, the choice of the target value, also known as regressand, which has to be predicted by sentence regression models is crucially important for building well-performing sentence regression models. Most of the recent works try to predict ROUGE recall scores of individual sentences, which seems to be a reasonable choice since the final summaries are also evaluated with ROUGE recall metrics (C.-Y. Lin, 2004; Owczarzak, Conroy, Dang, & Nenkova, 2012) (see Chapter 7 for more details). In Section 6.4.2, we explain that this choice may lead to suboptimal results and show in a wide range of experiments in Section 6.4.3 that our assumption is correct. We assume that the performance of current state-of-the-art sentence regression works can be improved by choosing a different regressand in the training phase.

6.4.1 Related Work

After the field of automatic summarization has been dominated by unsupervised extractive summarization models for some time (Carbonell & Goldstein, 1998; Erkan & Radev, 2004; S. Li, Ouyang, & Sun, 2006; Mihalcea & Tarau, 2004), supervised regression models are more commonly used in recent years.

Kupiec et al. (1995) proposed one of the first supervised summarization systems, which trains a Bayesian model to predict the probability that a sentence will be included in the summary. They criticized that although a large number of different features had been used in previous unsupervised models, no principled method to select or weight the features had been proposed at this time. Instead of generating summaries, the performance of the model was evaluated based on the classification output of the model for individual sentences. Similarly, Conroy and O'leary (2001) use a Hidden Markov Model to predict the probability that a sentence is included in a reference summary.

The model proposed by S. Li et al. (2006) predicts utility scores for individual sentences. The model weights are, however, not learned in a supervised training but assigned by humans. S. Li, Ouyang, Wang, and Sun (2007) extend this previously proposed unsupervised model and used a support vector regression (SVR) model in the DUC 2007 shared task (Over et al., 2007). Both S. Li et al. (2006) and S. Li et al. (2007) use a greedy selection strategy. Instead of learning to predict the probability of appearance of a sentence in a summary (Conroy & O'leary, 2001; Kupiec et al., 1995), S. Li et al. (2007) use the average and maximum text similarity of candidate sentences and reference summaries as regressands. (Ouyang, Li, Li, & Lu, 2011) also applied SVR but used the sum of word probabilities as regressand.

PriorSum (Cao, Wei, Li, et al., 2015) follows S. Li et al. (2007) and presents a linear regression framework which uses prior and document dependent features. As regressand, ROUGE$_2$ recall is used. Cao, Wei, Dong, Li, and Zhou (2015) propose a hierarchical regression process that predicts the importance of sentences based on their constituents. ROUGE$_1$ recall and ROUGE$_2$ recall are used as regressand for sentences. For sentence selection, they implement a greedy selection and a selection based on integer linear programming. The Redundancy-Aware Sentence Regression (Ren, Wei, & Chen, 2016) framework models both importance and redundancy jointly. They train a multi-layer perceptron which then predicts relative importance utilities based on ROUGE$_2$ recall scores. REGSUM (Hong et al., 2014) predicts sentence importance based on word importance and additional features. They use a greedy selection strategy with additional redundancy avoidance, which only appends sentences to the summary if the maximum cosine similarity to already selected sentences is lower than a fixed threshold.

ROUGE recall is often used in sentence regression in combination with a greedy selection and an additional redundancy avoidance strategy. In the following, we first describe the underlying intuition of using ROUGE recall. Second, we describe why using ROUGE precision instead can be potentially better. Later, we show in the experiments that using ROUGE precision is not only theoretically appealing but also works better in practice than ROUGE recall.

6.4.2 ROUGE Recall vs. ROUGE Precision

ROUGE (C.-Y. Lin, 2004) is the method of choice for the evaluation of generated summaries in the field of automatic summarization. Its idea is to compute the similarity between automatically generated summaries and reference summaries, which are typically provided by humans. ROUGE can be viewed as an evaluation measure for an information retrieval task in which precision and recall can be measured. Let E be a set of elements (for example, E denotes all unigrams in ROUGE$_1$ and all bigrams in ROUGE$_2$), $\zeta(X) \subset E$ the multiset of desired elements in reference summary X, $\zeta(\hat{X}) \subset E$ the multiset of elements in the automatically produced summary $\hat{X}$, and $|.|$ the size of a multiset. The recall of $\zeta(\hat{X})$ with respect to $\zeta(X)$ is defined as

$$r(\zeta(\hat{X}), \zeta(X)) = \frac{|\zeta(\hat{X}) \cap \zeta(X)|}{|\zeta(X)|}, \tag{6.7}$$

where the intersection $\cap$ of two multisets is defined as the smallest multiset S with

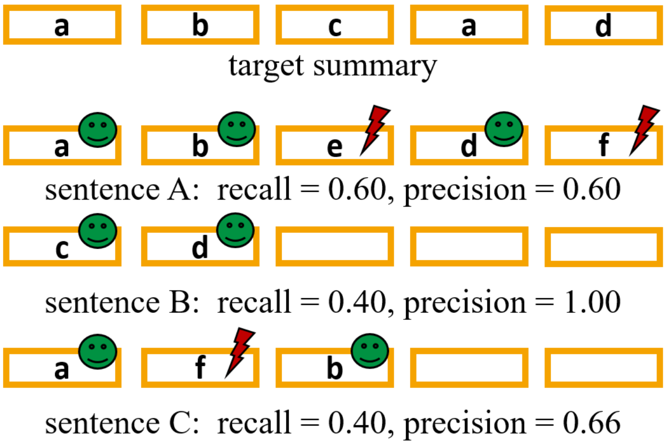
$$\mathrm{num}_S(e) = \min(\mathrm{num}_{\zeta(\hat{X})}(e), \mathrm{num}_{\zeta(X)}(e)) \ \forall e \in \zeta(\hat{X}), \zeta(X), \tag{6.8}$$

where $\mathrm{num}_S(e)$ indicates the number of appearances of element e in multiset S.

The recall measures how much of the desired content in $\zeta(X)$ has been returned by the system which created $\zeta(\hat{X})$. On the other hand, precision is defined as

$$p(\zeta(\hat{X}), \zeta(X)) = \frac{|\zeta(\hat{X}) \cap \zeta(X)|}{|\zeta(\hat{X})|}, \tag{6.9}$$

and measures how much of the returned content in $\hat{X}$ was actually desirable according to there reference summary X.



a
b
c
a
d
target summary
a
b
e
d
f
sentence A: recall = 0.60, precision = 0.60
c
d
sentence B: recall = 0.40, precision = 1.00
a
f
b
sentence C: recall = 0.40, precision = 0.66

Figure 6.4.: Exemplary illustration of selecting sentences according to precision and recall. The target summary has five slots. Sentence A will be selected according to recall since it has a recall scores of 0.6, whereas sentence B and C only have a recall score of 0.4. Sentence A, however, occupies already all available slots in the summary. No more sentence can be selected. Sentence B will be first selected according to precision due to a precision score of 1.0. After the selection of sentence B, three slots are still available in the summary which can be used to fit sentence C to improve the overall summary recall to 0.8.

In ROUGE$_n$, the multiset E is defined as the set of all n-grams, the desired reference multiset $\zeta(X)$ contains all n-grams in a reference summary, and the multiset $\zeta(\hat{X})$ contains all n-grams in the system summary. We use multisets and not sets since the same n-gram can be contained multiple times in a text.

When ROUGE was first introduced as the evaluation metric for the DUC 2003 shared task (Over et al., 2007), C.-Y. Lin and Hovy (2003) report that metrics based on ROUGE recall scores have a good agreement with human judgments. A summary with a high ROUGE recall contains many n-grams which also appear in the reference summaries. Owczarzak et al. (2012) show that ROUGE$_2$ recall has the highest agreement with human judgments if automatically generated summaries have to be evaluated. ROUGE$_2$ recall is therefore often used to evaluate automatic summarization systems.

Crucial for the use of ROUGE recall is the length limitation of the generated summaries. Usually, the generated summaries are limited to a fixed number of words or characters. Without such a length restriction, systems would be able to generate arbitrary long texts to increase the recall.

Summarization systems aim at maximizing ROUGE recall scores of the generated summaries since the final summaries are usually evaluated with ROUGE recall. Most greedy extractive summarization approaches try to maximize the overall ROUGE recall of a summary by incrementally adding sentences with a high ROUGE recall to the summary. The idea of this strategy is to pack as much important content as possible into the summary in each greedy step in order to increase the ROUGE recall of the resulting summary as much as possible. What is usually not considered is the fact that this strategy tends to select longer sentences, since longer sentences tend to have a higher recall. As a result, fewer sentences can be selected since the maximum length of the summary is reached earlier.

An alternative strategy, which has not been discussed in the literature so far, is to select sentences according to their ROUGE precision scores. The idea behind this approach is not to cover as much information as possible in every step but to waste as little space as possible in every step. Selecting sentences according to precision will not have a bias for longer sentences but short and dense sentences. Since this strategy tends to selected shorter sentences, more sentences can be included in the summary, which can, in turn, result in a higher overall ROUGE recall of the resulting summary. Figure 6.4 shows an example in which selecting sentences according to ROUGE precision leads to a higher ROUGE recall score of the resulting summary than selecting sentences according to ROUGE recall.

We summarize that selecting sentences according to ROUGE precision scores can, intuitively, be better than selecting sentences according to ROUGE recall scores even though the final summaries are evaluated with ROUGE recall metrics. In the following section, we show that this intuition is not only appealing in theory but can also be substantiated in empirical experiments.

6.4.3 Experiments

In this section, we present the experimental setup in which we test different regressand candidates for sentence regression in three different, well-known multi-document summarization corpora. We used the corpora from the DUC 2004, TAC 2008, and TAC 2009 summarization shared tasks (see Chapter 5 for details). We simulate in the experiments the outcomes of regression models that use different regressands. These experiments provide us with insights on which regressand candidates should be considered in regression models. For our experiments, we produce summaries containing 665 characters for DUC2004 and summaries containing 100 words for TAC2008 and TAC2009 following the standard length choices.

Regressand Candidates

We examine seven different regressand candidates (**in boldface**), which can be used as regressands when the utility function u is learned via supervised regression.

ROUGE$_1$ recall (**R1 Rec**) and ROUGE$_2$ recall (**R2 Rec**) are computed according to Equation 6.7 for all sentences in the input documents. ROUGE$_n$ recall counts the n-gram overlap of the input sentence and the reference summaries. The more n-grams in the reference documents are covered by a sentence, the higher the score is. These regressands are usually used by prior sentence regression works.

We also compute the ROUGE$_1$ precision (**R1 Prec**) and ROUGE$_2$ precision (**R2 Prec**) for all sentences according to Equation 6.9. A sentence has a high ROUGE$_n$ precision if a high rate of n-grams in the sentence match with n-grams in the reference documents. Sentences with a high density of matching n-grams are therefore preferred by ROUGE precision. Based on the argumentation in the previous section, we suspect that R1 Prec and R2 Prec perform better than R1 Rec and R2 Rec.

As a reference point, we compute for each sentence the maximum similarity (**maxADW**) for and the average similarity (**avgADW**) with all sentences in the reference summaries according to the state-of-the-art

	avg. stems			avg. sentences		
	D04	T08	T09	D04	T08	T09
R1 Rec	166	132	141	3.42	2.67	2.70
R2 Rec	160	129	132	4.26	3.46	3.55
R1 Prec	157	125	127	7.76	6.75	6.07
R2 Prec	157	129	126	7.10	6.13	6.09
maxADW	158	127	129	6.56	5.06	5.11
avgADW	158	126	126	5.12	4.13	4.02
random	164	131	131	6.66	5.21	4.89

Table 6.8.: Averaged lengths of resulting summaries measured in number of stems (avg. stems) and number of sentences (avg. sentences). D04 refers to DUC2004 and T08 and T09 refer to TAC2008 and TAC2009, respectively. We count also partially contained sentences which have been cut by the ROUGE length limitation.

Align-Disambiguate-Walk (ADW) similarity measure (Pilehvar, Jurgens, & Navigli, 2013). ADW computes the semantic similarity of two sentences by finding an optimal alignment of word senses contained in the two sentences.

For a given sentence **s** and a reference summary X, maxADW is computed by

$$\text{maxADW}(\mathbf{s}) = \max_{\mathbf{t}\in X} \text{ADWsim}(\mathbf{s},\mathbf{t}) \tag{6.10}$$

and avgADW is computed by

$$\text{avgADW}(\mathbf{s}) = \frac{1}{|X|} \cdot \sum_{\mathbf{t}\in X} \text{ADWsim}(\mathbf{s},\mathbf{t}). \tag{6.11}$$

Computing the maximum similarity follows the idea that a good sentence in the input documents matches well with one sentence in the reference summary. A sentence is representative of the whole summary if it has a high average similarity with all the reference summary sentences. For each sentence, we also randomly generated (**random**) sentence scores, which are used as regressand.

Length Analysis

Table 6.8 provides details about the lengths of the produced summaries according to the number of stems and number of sentences. The hypothesis that an algorithm that selects sentences according to recall tends to select longer sentences is confirmed. Hence, the results also confirm that longer sentences tend to have higher recall scores.

Optimal Score Prediction without Redundancy Avoidance

In the next experiment, we investigate how helpful the predicted scores are under the assumption that the regressands can be predicted perfectly. The experiment evaluates how a system performs in the

	DUC2004		TAC2008		TAC2009	
	R-1	R-2	R-1	R-2	R-1	R-2
R1 Rec	38.63	08.99	39.28	11.08	34.31	08.37
R2 Rec	39.23	12.07	42.39	16.20	37.42	13.03
R1 Prec	**41.29**	11.18	**43.56**	14.65	**39.45**	12.17
R2 Prec	39.18	**12.73**	43.46	**18.19**	37.81	**13.64**
maxADW	37.60	10.13	42.55	15.46	34.56	11.05
avgADW	38.50	09.62	40.97	12.43	35.48	09.34
random	31.76	04.66	29.58	04.60	29.88	04.63

Table 6.9.: Summarization results in three different multi-document summarization corpora without redundancy avoidance. Columns R-1 and R-2 display the summary quality according to ROUGE$_1$ recall and ROUGE$_2$ recall scores, respectively.

optimal case. We do not consider redundancy avoidance strategies in this experiment so that observed performance differences are solely due to differences in the used regressand candidates.

The results of the experiment are shown in Table 6.9. It can be seen that in all corpora the use of ROUGE$_1$ precision regressands of the sentences leads to better results than using ROUGE$_1$ recall regressands if ROUGE$_1$ recall is used as an evaluation metric for the final summary. Analogous results can be observed for ROUGE$_2$ scores. These results indicate that using ROUGE recall as regressand in a sentence regression framework is not very promising. Thus, the results are a first confirmation of the previously described intuition that predicting precision scores can be lead to better results than predicting recall scores.

German Data

In addition to the standard DUC and TAC corpora, we also report results for 2 German datasets, namely the DBS corpus (Benikova et al., 2016) and a subset of the German part of the auto-*h*MDS corpus (Zopf, 2018a; Zopf, Peyrard, & Eckle-Kohler, 2016). The DBS corpus contains topics from the educational domain. auto-*h*MDS contains heterogeneous topics retrieved from Wikipedia and automatically collected source documents retrieved from web sites.[6] The results are displayed in Table 6.10 and show that the previously observed results in English corpora can also be observed in German datasets. We additionally observe that ROUGE$_1$ precision seems to be a bit stronger in DBS compared to ROUGE$_2$ precision even if the resulting summaries are evaluated with ROUGE$_2$ recall.

Optimal Prediction of F-Scores

The previous experiment clearly showed that selecting sentences according to ROUGE precision outperforms a selection according to ROUGE recall. In the next experiment, we evaluate if a trade-off between recall and precision can lead to even better results. It is known that in inductive rule learning, parametrized measures such as the *m*-estimate, which may be viewed as a trade-off between precision and weighted relative accuracy, can be tuned to outperform its constituent heuristics (Janssen &

[6] See Chapter 5 for details about auto-*h*MDS.

	DBS		hMDS	
	R-1	R-2	R-1	R-2
R1 Rec	33.48	13.89	31.94	13.38
R2 Rec	38.67	21.77	40.67	24.39
R1 Prec	**42.20**	**25.55**	**43.25**	23.01
R2 Prec	37.01	23.12	41.65	**24.96**
random	23.27	04.23	20.63	02.36

Table 6.10.: Results as in Table 6.9, but for 2 datasets (DBS and auto-hMDS) containing German documents.

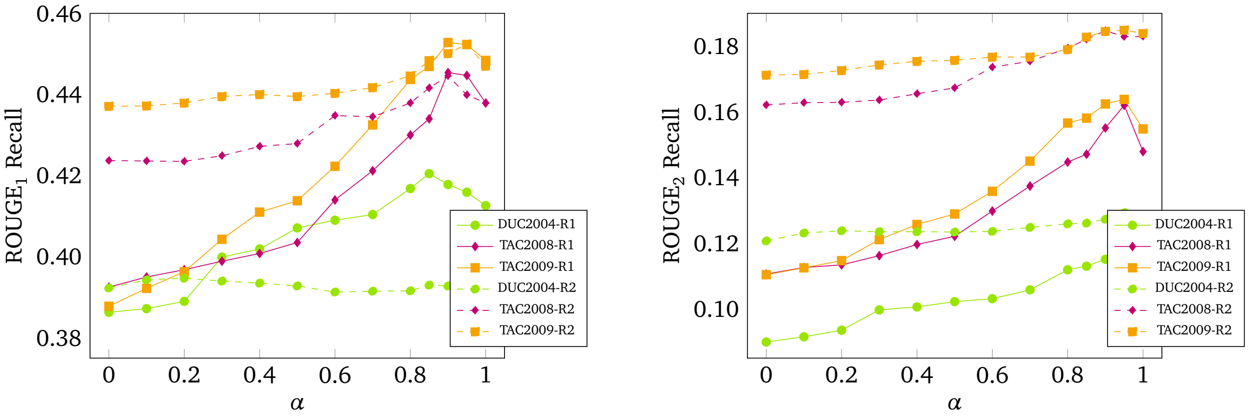


Figure 6.5.: Results of mixing ROUGE$_{1/2}$ precision and ROUGE$_{1/2}$ recall using $F_\alpha(p,r)$-Measure in different datasets evaluated with ROUGE$_1$ recall (left) and ROUGE$_2$ recall (right). For example, the curve labeled DUC2004-R1 shows the results of mixing ROUGE$_1$ precision and ROUGE$_1$ recall in the DUC 2004 corpus.

Fürnkranz, 2010). In retrieval tasks, the F-measure provides a more commonly used trade-off between precision and recall, so we chose to use this measure for our experiments. We compute for all sentences the F-measure with $0 \leq \alpha \leq 1$ as

$$F_\alpha(p,r) = \frac{1}{\frac{\alpha}{p} + \frac{1-\alpha}{r}} \tag{6.12}$$

where a $F_0(p,r)$ is equivalent to recall and $F_1(p,r)$ equals precision.

The results of the experiment, which are displayed in Figure 6.5, show that precision ($\alpha = 1.0$) is already close to the optimum but that incorporating also a small fraction of recall ($\alpha \approx 0.9$) leads to the best results which indicates that a slight bias towards longer sentences can improve the result even further. A possible explanation is that there are short sentences in the input documents which are considered redundant to other high precision sentences. However, overall the trend in the results (increasing evaluation scores with increasing α, which means increasing impact of ROUGE precision) substantiate the general hypothesis that sentence regression works may consider ROUGE precision instead of ROUGE recall as regressand.

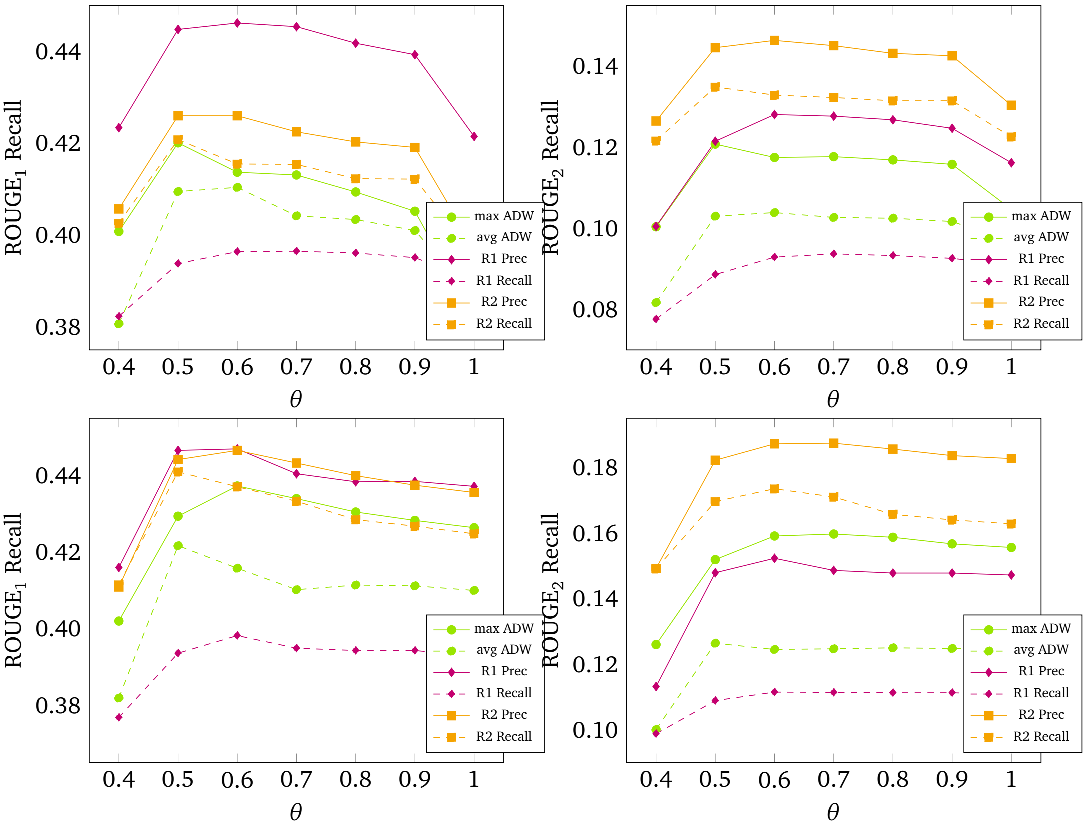


Figure 6.6.: Summary quality assessed with ROUGE$_1$ recall and ROUGE$_2$ recall with different redundancy avoidance thresholds θ in the DUC 2004 (top half) and TAC 2008 (bottom half) datasets.

Optimal Prediction with Redundancy Avoidance

Summarization systems usually apply a redundancy avoidance strategy in order to avoid including the same information multiple times in the summary. In this experiment, we investigate whether incorporating a simple redundancy avoidance strategy will lead to different results.

During the greedy selection process, we compute the similarity of the currently highest scoring sentence and all already selected sentences (see Algorithm 3). The highest scoring sentence is skipped if the maximum similarity of the sentence and the already selected sentences is higher than a predefined threshold θ. We use the state-of-the-art ADW similarity measure to compute the similarities and test the quality of the generated summaries as in the previous experiments with ROUGE$_1$ and ROUGE$_2$ recall. The results of the experiment for the thresholds $\theta = 0.4, 0.5, \ldots, 1.0$ are displayed in Figure 6.6.

We see that sentence selection using ROUGE$_{1/2}$ precision scores (solid red and solid blue lines) consistently leads to better results than ROUGE$_{1/2}$ recall scores (red and blue dashed lines) for all chosen redundancy thresholds. Selecting according to maximum ADW similarity leads to consistently better results than selecting according to the average ADW similarity, which indicates that it is better to search for sentences that align well with a part of the summary than selecting sentences that align relatively well with the whole summary. The best results are achieved with thresholds of $\theta = 0.5$ and $\theta = 0.6$, which work well for ROUGE$_1$ and ROUGE$_2$ recall in both datasets.

	score	DUC2004		TAC2008		TAC2009	
		R-1	R-2	R-1	R-2	R-1	R-2
$\mathcal{U}$(-0.2, 0.2)	R1 Rec	37.22	07.71	36.73	08.79	37.06	08.99
	R2 Rec	36.93	08.74	36.45	09.91	37.83	11.06
	R1 Prec	**42.53**	10.87	**42.19**	12.57	**43.65**	13.58
	R2 Prec	40.37	**12.04**	40.63	**14.23**	42.25	**15.49**
$\mathcal{U}$(-0.3, 0.3)	R1 Rec	36.78	07.43	35.70	08.00	36.04	08.27
	R2 Rec	35.45	07.54	34.62	08.58	36.08	09.43
	R1 Prec	**42.02**	10.45	**41.42**	11.75	**42.75**	12.83
	R2 Prec	39.56	**11.16**	38.94	**12.64**	40.91	**14.29**
$\mathcal{U}$(-0.4, 0.4)	R1 Rec	36.10	06.92	34.91	07.48	35.85	07.93
	R2 Rec	34.92	07.32	34.08	07.85	3.545	08.70
	R1 Prec	**41.27**	09.98	**40.44**	11.04	**41.63**	11.92
	R2 Prec	39.02	**10.63**	38.22	**11.74**	39.51	**12.97**

Table 6.11.: Summarization results in three different multi-document summarization corpora with noisy score prediction with uniform noise.

	score	DUC2004		TAC2008		TAC2009	
		R-1	R-2	R-1	R-2	R-1	R-2
$\mathcal{N}$(0, 0.05)	R1 Rec	37.53	07.93	36.99	09.31	37.40	09.36
	R2 Rec	35.46	07.60	35.50	09.41	36.07	09.96
	R1 Prec	**43.55**	11.99	**43.59**	13.98	**45.58**	15.56
	R2 Prec	41.06	**12.92**	42.80	**16.46**	43.97	**17.48**
$\mathcal{N}$(0, 0.1)	R1 Rec	35.63	06.83	34.45	07.31	35.06	07.57
	R2 Rec	33.39	06.04	32.76	06.93	32.88	07.98
	R1 Prec	**41.70**	10.19	**41.41**	12.09	**43.06**	13.23
	R2 Prec	38.41	**10.33**	38.27	**12.43**	40.15	**13.94**
$\mathcal{N}$(0, 0.2)	R1 Rec	33.59	05.72	32.00	05.78	32.36	05.99
	R2 Rec	32.64	05.28	30.76	05.48	31.47	06.01
	R1 Prec	**38.19**	**08.01**	**37.34**	**09.00**	**38.75**	**10.06**
	R2 Prec	35.07	07.45	34.08	08.45	34.71	09.08

Table 6.12.: Summarization results in three different multi-document summarization corpora with noisy score prediction with Gaussian noise.

Noisy Predictions

In the previous experiments, we showed the results of a greedy summarizer which selects sentences according to perfectly predicted scores. Summarization systems are, however, not capable of predicting the scores perfectly. Hence, we investigate whether imperfect predictions have an influence on our results in the next experiment. This investigation will also provide insights into the robustness of a greedy summarizer in the presence of imprecise predictions.

In order to get model-independent results, we simulate imperfect precisions by adding two different kinds of noise to simulate imperfect predictions, namely additive uniformly distributed continuous noise $\mathcal{U}(a,b)$ and additive Gaussian noise $\mathcal{N}(\mu,\sigma^2)$. For the uniform noise $\mathcal{U}(a,b)$, we test boundaries from $a = -0.2$, $b = 0.2$ to $a = -0.4$, $b = 0.4$. For Gaussian noise, we use mean $\mu = 0$ and variance

$\sigma^2 \in \{0.05, 0.1, 0.2\}$. Based on the results in the previous section, we fix the redundancy threshold to 0.6 in this experiment. Due to the random noise, the experiments are no longer deterministic. Hence, we run each experiment 10 times and report averaged results.

The results of these experiments (see Table 6.11 and Table 6.12) confirm that predicting ROUGE precision is always better than predicting ROUGE recall, in the presence of different kinds of noises and different noise intensities. In case strong Gaussian noise is applied (Table 6.12, last block), the quality of the summaries decreases more strongly if ROUGE$_2$ precision scores are predicted, which means that predicting ROUGE$_1$ precision might be better than predicting ROUGE$_2$ precision in the case of low prediction quality.

6.4.4 Conclusions

Current state-of-the-art sentence regression systems for automatic summarization learn to predict ROUGE recall scores of individual sentences and apply a greedy sentence selection strategy in order to generate summaries. We show in a wide range of experiments that this design choice leads to suboptimal results. In all experiments, we observed the same pattern. The resulting summaries will have a lower quality if ROUGE recall scores for sentences are used instead of ROUGE precision - no matter whether or not redundancy avoidance is considered and whether or not the scores can be predicted perfectly.

In an experiment where we combined both ROUGE recall and ROUGE precision with an F-score computation, we confirmed the previously described observation that the quality of summaries tends to improve with a growing ratio of ROUGE precision vs. ROUGE recall, with a maximum performance for a ratio of $\alpha \approx 0.9$. Hence, biasing the greedy sentence selection slightly towards longer sentences is promising. This result goes in line with an often applied pre-processing step in which very short sentences are discarded without further analysis (Cao, Wei, Li, et al., 2015; Erkan & Radev, 2004).

We also presented an intuition why a selection according to ROUGE precision leads to better results. A system that selects according to ROUGE recall will tend to select longer sentences since longer sentences tend to have a higher recall. We conclude that systems should instead of iteratively fitting as much as possible into a summary rather aim at wasting as little space as possible in every step.

For future works, it is straightforward to incorporate these findings. Instead of learning to predict ROUGE recall scores, the regressand can simply be exchanged, and the ROUGE precision can be used instead. We expect that summarization models can benefit from this modification. We furthermore conclude that comparisons between ILP and greedy methods (Cao, Wei, Dong, et al., 2015) are biased in favor of ILP. A better comparison is possible if precision scores are used as input for greedy systems instead of recall scores.

6.5 Summary

In this chapter, we discussed how a context-free information importance estimator can be built with machine learning. Using machine learning instead of classical computer programming is necessary if it is not possible for humans to write algorithms that solve a problem directly. For some problems such as multi-document summarization, there is not enough training data available to train supervised machine learning models. An alternative is to use incidental supervision to learn signals for importance estimation from unlabeled data.

Prior works in automatic summarization mainly rely on importance signals derived from the source documents, which makes sense if it is known that the source documents can be used to generate such signals. Source documents in newswire datasets, for example, have been written by journalists with a particular purpose with a particular style and can, therefore, be used to extract said signals.

Assumptions that can reasonably be made for newswire documents are not necessarily met in heterogeneous summarization in which the source documents have not been written with a particular purpose. Hence, importance signals can not reliably be extracted from heterogeneous sources.

To summarize heterogeneous source documents successfully, summarization systems cannot rely on signals derived from the source document at hand. Instead, summarization systems have to learn prior knowledge to be able to estimate information importance without document-derived importance signals. We proposed CPSum, a simple preference-based learning model that learns to predict the probability that information is promoted to a summary based on the information contained in the source documents. Since CPSum does not use document-derived features, it is a context-free importance estimator according to Definition 22. CPSum is robust against modifications to the data such as information order randomization and information oversampling.

We have furthermore investigated which regressands should be used in sentence regression frameworks. We observe that ROUGE recall scores are usually used as target values for the regression. The experiments we conducted show that this design choice may lead to suboptimal results. Instead of ROUGE recall, it is better to use the ROUGE precision score. We also provide an intuition why this makes sense. Ordering sentences according to ROUGE precision allows a greedy sentence selection algorithm to waste as little space as possible in each step instead of fitting as much content as possible into the summary which is the case if ROUGE recall is used. Our experiments also show that comparisons in recent studies between integer linear programming (ILP) and greedy systems are biased in favor of ILP systems since sentences are ordered in both cases according to ROUGE recall scores. This decision is an optimal choice for ILP systems, but a suboptimal choice for greedy methods. A fair comparison has to sort the sentences according to ROUGE precision scores for the greedy selection.

Sentence regression is appealing for context-free information importance estimation since a lot of training data is available in single-document summarization datasets to train sentence regression models. Context-free sentence regression models can be trained in SDS dataset since features based on information frequency, which are usually not strong in single-document summarization dataset, are not used by context-free models.

7 Learning to Evaluate Automatic Summarization with Context-free Pairwise Preferences

The third major ingredient necessary to develop machine learning models is a proper evaluation methodology. Hence, we focus in this chapter on the question: **How can information importance estimators be evaluated automatically?**

We start in Section 7.1 by discussing related foundations. Contrary to many machine learning tasks, the output space in automatic summarization is very big and an assessment of machine outputs according to labels such as *correct* and *incorrect* is not practical. Hence, evaluation in automatic summarization requires a gradually assessment of produced summary quality.

We review prior work in Section 7.2 which includes automatic sentence-level evaluation (Section 7.2.1) manual summary-level evaluation such as SEE and the Pyramid method (Section 7.2.2) and prior work for automatic summary-level evaluation such as ROUGE (Section 7.2.3). In Section 7.2.4, we briefly discuss manual and automatic evaluation in general.

The performance of evaluation models has to be evaluated just like the performance of all machine learning models before they can be used in practice. Hence, we discuss the evaluation of evaluation models in Section 7.3 in detail. In particular, we describe the evaluation which has been used in the past in Section 7.3.1 and discuss limitations in Section 7.3.2. We provide an counter-example with poor performance but very high correlation in Section 7.3.3. Finally, we discuss a more sound way to evaluate evaluation models in Section 7.3.4.

A significant limitation of many evaluation methods is that they require the availability of multiple human-written summaries to asses the quality of automatically generated summaries. The creation of reference summaries is, however, expensive and complicated, as already discussed previously. Furthermore, these models require a reliable estimation of semantic similarity of texts is an unsolved problem so far. Hence, we present a new evaluation method which does not require reference summaries and does not compare texts in Section 7.4. Contrary to prior works, we use cheap pairwise preferences between sentences to estimate the quality of summaries. We describe the model in detail in Section 7.4.1 and describe three ways to obtain pairwise preferences in Section 7.4.2. The evaluation in Section 7.4.3 shows that the newly proposed model requires less annotation effort and performs better than the state-of-the-art. We conclude in Section 7.4.4.

We summarize this chapter in Section 7.5.

7.1 Foundations

Chapter 5 and Chapter 6 discussed the first two essential part for machine learning, namely datasets and learning algorithms. We provide in this section the foundations for the last part, namely *evaluation*. To

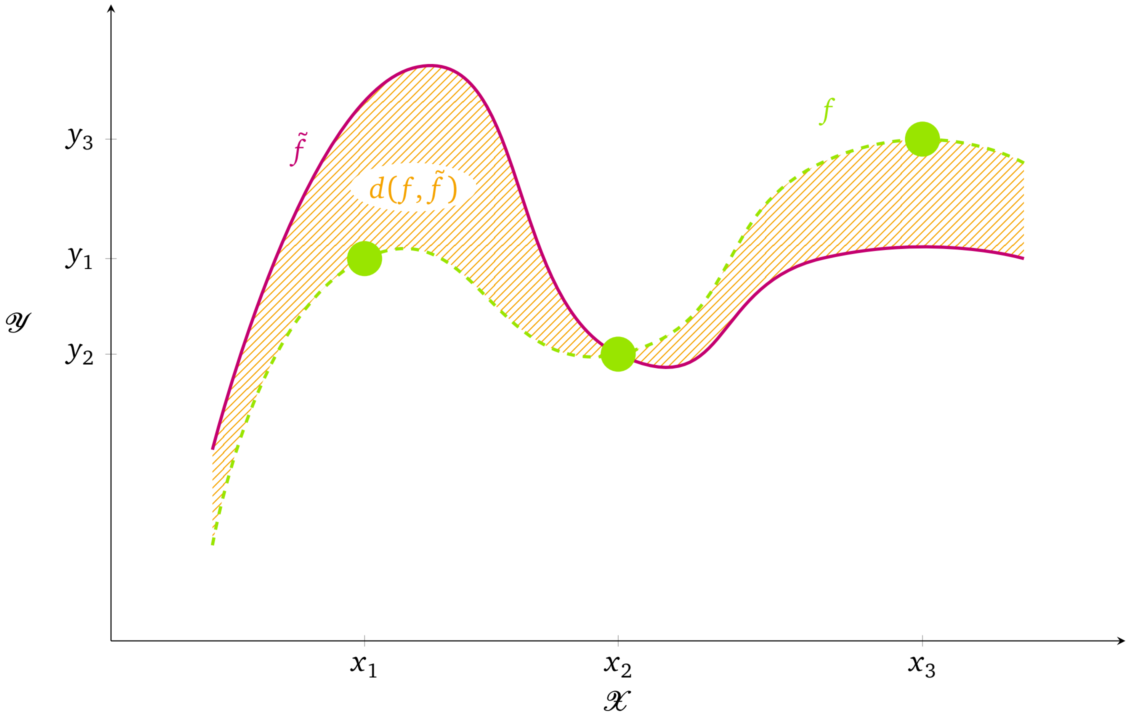


Figure 7.1.: Illustration of evaluation in machine learning. The visualization extends Figure 6.1. Given two functions f and $\tilde{f}$, evaluation aims at estimating the distance d between f and $\tilde{f}$.

this end, we extend the terminology introduced in Section 5.1 which has already been extended once in Section 6 one more time. In previous sections, we defined a task as function $f : \mathscr{X} \rightarrow \mathscr{Y}$ mapping from domain $\mathscr{X}$ to codomain $\mathscr{Y}$ and discussed how functions $\tilde{f}$ can be learned which which aim at approximating the given function f.

Evaluation aims at estimating how well the model $\tilde{f}$ learned by a machine learning algorithm can perform the task f. In other words, evaluation aims at estimating how big the *distance* between $\tilde{f}$ and f is. The distance between the learned approximation $\tilde{f}$ and f can be computed by calculating the distance between the predicted output and the correct output for all possible inputs in $\mathscr{X}$. Let $d(\mathbf{X}_i) = |f(\mathbf{X}_i) - \tilde{f}(\mathbf{X}_i))|$ denote the point-wise distance between f and $\tilde{f}$. The distance between the functions can be computed by summing all distances for all input points. Hence, $\int_{\mathbf{X}} |f(\mathbf{X}_i) - \tilde{f}(\mathbf{X}_i))|$ can be used to calculate the distance between $\tilde{f}$ and f. The distance is illustrated by the area between $\tilde{f}$ and f in Figure 7.1.

Recall that datasets usually only contain a subset of all input-output pairs from the task at hand specified by f. Therefore, the distance between the two functions can not be calculated as described above. Hence, only approximations of the true distance can be calculated. We call an approximation of the distance d *error* and denote the error by e.

Training and Test Sets

Finding an approximation of f for a finite set of known input-output pairs in the training data $D_{\text{train}} = (\mathbf{X}_1, \mathbf{Y}_1), \ldots, (\mathbf{X}_n, \mathbf{Y}_n)$ can easily be achieved by storing all observed pairs. Hence, the error e on observed samples can be minimized easily. This, however, does not necessarily provide a good insight on the true distance d between the learned model and the task at hand since the approximation can be different for not observed pairs $(\mathbf{X}_i, \mathbf{Y}_i) \notin D_{\text{train}}$. To achieve a good approximation of d, the dataset can be splitted into two different subsets, a training set D_{train} and a test set D_{test} with $D = D_{\text{train}} \cup D_{\text{test}}$ and $D_{\text{train}} \cap D_{\text{test}} = \emptyset$. D_{train} can be used as described before to learn the function $\tilde{f}$ whereas D_{test} remains hidden during the training process and is only used for evaluation. Even if the model remembers all seen input-output pairs it is not guaranteed that $\tilde{f}$ will also make correct predictions on unseen data points in D_{test}. Calculating e on unseen data therefore provides a better insight on the true distance d. The error e calculated on D_{train} is called *training error* and the error e calculated on the unseen test dataset D_{test} is called *test error*. We use e_{train} and e_{test} to distinguish between training and test error if necessary.

Accuracy, Precision, and Recall

Extractive summarization is closely related to information retrieval (IR) where sets of instances (sentences, for example) have to be labeled as either *relevant* or *not relevant*.

Definition 24 (Positive and Negative Instances). Let x_i be a set of instances (e.g., sentences stemming from the i-th text in a dataset), p_i the sentences which have been labeled as relevant, and n_i the sentences which have been labeled as not relevant. Instances in p_i are called **positive instances** and instances in n_i are called **negative instances**.

Similarly, extractive summarization systems can label sentences as relevant and as not relevant yielding sets $\tilde{p_i}$ and $\tilde{n_i}$, respectively. Since sentences cannot be labeled as relevant and as not relevant at the same time by humans or machines, we obtain $\emptyset = p_i \cap n_i = \tilde{p_i} \cap \tilde{n_i}$ and $x_i = p_i \cup n_i = \tilde{p_i} \cup \tilde{n_i}$.

Definition 25 (True Positives, False Positives, True Negatives, and False Negatives). The elements in set $tp_i = p_i \cap \tilde{p_i}$ are called **true positives**, the elements in set $fp_i = n_i \cap \tilde{p_i}$ are called **false positives**, the elements in set $tn_i = n_i \cap \tilde{n_i}$ are called **true negatives**, and the elements in set $fn_i = p_i \cap \tilde{n_i}$ are called **false negatives**.

The sets defined above can be used to evaluate information retrieval systems that label instances according to their relevance. The accuracy defined as follows.

Definition 26 (Accuracy). Let tp, tn, fp and fn be as defined in Definition 25. The **accuracy** is defined as

$$\text{accuracy} = \frac{|tp| + |tn|}{|tp| + |tn| + |fp| + |fn|} \tag{7.1}$$

The precision indicates the fraction of instances that have been correctly labeled positive among all instances which have been labeled positive. Hence, the precision indicates how many instances which have been labeled as positive are incorrectly labeled.

Definition 27 (Accuracy). Let tp and fp be as defined in Definition 25. The **precision** is defined as

$$\text{precision} = \frac{|tp|}{|tp| + |fp|} \tag{7.2}$$

The recall indicates the fraction of instances that have been correctly labeled positive among all positive instances.

Definition 28 (Recall). Let tp and fn be as defined in Definition 25. The **recall** is defined as

$$\text{recall} = \frac{|tp|}{|tp| + |fn|} \tag{7.3}$$

On the Hardness of Evaluation

In many classical machine learning problems, the output space $\mathscr{Y}$ is simple. For instance, the output space in binary classification contains only two possible values, usually called positive and negative class. Hence, $\mathscr{Y} = \{\text{positive}, \text{negative}\}$ and $|\mathscr{Y}| = 2$. For tasks with small output spaces, it is rather simple to evaluate machine learning models since the error can be defined for each possible confusion manually. In binary classification, for example, the error of predicting the class as positive while the true class is negative (i.e., a false positive) can be set to a task-specific value e_{fp}. Similarly, the error for each false negative can be set to e_{fn}. Based on the two errors, the performance of a model can be computed automatically, for example, on all pairs in a test dataset D_{test}). Errors can also be automatically computed in regression tasks. Regression task have a subset of $\mathbb{R}$ as output space which allows the calculation of errors such as absolute error, which is defined as $|\tilde{h} - f|$, or a squared error, which is defined as $(\tilde{h} - f)^2$. The sum of the absolute or squared error over all input-output pairs in a test dataset can then be used to evaluate the model $\tilde{h}$.

In summarization, the output space $\mathscr{Y}$ contains all possible texts since summarization systems are free to produce any text as output. It is unfortunately not possible to automatically compute a distance easily in this space. Hence, estimating how good a particular summary is can not be performed similarly to regression tasks. The output space is furthermore not restricted to a few possible outcomes. Even if a length restriction limits the number of possible outputs, the output space is still huge. Hence, asking humans to assess the quality of all possible outputs is infeasible. Furthermore, multiple different elements in $\mathscr{Y}$ may be considered to be a correct output. Two summaries that use different wording but contain the same content, for example, might be considered to be equally good. The quality of a summary can also be estimated differently by multiple people since different people estimate the importance of information differently, as discussed in Chapter 2. Hence, evaluation in automatic summarization is a difficult problem.

In general, correct evaluation is even harder than correct output prediction. This can easily be shown as follows. If the distance d is known for every possible task input $\mathbf{X}_i \in \mathscr{X}$, we can find the correct $\mathbf{Y}_i$ for every $\mathbf{X}_i$ by searching for $\mathbf{Y}_i \in \mathscr{Y}$ with $\mathbf{Y}_i = \arg\min d(f(\mathbf{X}_i), \tilde{f}(\mathbf{X}_i))$. This might be computationally expensive due to a possibly (infinitively) large search space or a slow computation of d, but is still possible in theory. Hence, we can find a function $\tilde{f}$ such that $\int_X d(f, \tilde{f}) \leq \int_X d(h, \tilde{f}) \forall h$. On the other hand, if the correct f is known, nothing can be inferred in general about the distance $d(f, h)$ for other functions $h \neq f$. Hence, the prediction problem is solved if the evaluation problem is solve, but the evaluation problem is not solved if the prediction the is solved.

Machine Learning for Evaluation

In Chapter 6, we discussed that writing algorithms manually for summarization is a problematic endeavor. This problem motivates the use of machine learning for automatic summarization. Similarly, we can also train evaluation models with machine learning to estimate the quality of generated summaries.

Since learning to evaluate the performance of a model is a difficult task, evaluation models require additional data to perform well. We call this data *evaluation data* since it is used by evaluation models for evaluation. Evaluation data is not part of the original dataset which contains only input-output pairs $(\mathbf{X}_i, \mathbf{Y}_i)$ but supplements the dataset with additional information $\mathbf{Z}_i$. We call datasets with supplementary data *automatically evaluable dataset*. Automatically evaluable dataset are not tuples of input-output pairs $(\mathbf{X}_i, \mathbf{Y}_i)$ but triples $(\mathbf{X}_i, \mathbf{Y}_i, \mathbf{Z}_i)$.

We discuss in upcoming Section 7.2 different types of supplementary evaluation data $\mathbf{Z}_i$.

Statistical Tests

Statistical significance tests can be used to calculate how likely it is that an observed effect in dataset $\mathbf{X}_1 \subset \mathscr{X}$ has occurred due to a sampling error alone and that the observation would not be observable in another sample of the data (i.e., by using another set of samples $\mathbf{X}_2 \subset \mathscr{X}$). Applying statistical significance tests requires, however, that the samples in $\mathbf{D}_1$ have been drawn independently and identically distributed from $\mathscr{X}$. This prerequisite is usually not satisfied in datasets modeling real-world problems such as summarization, which renders significance tests unsuitable in these situations. Hence, we opt for not performing significance tests.

7.2 Evaluation in Automatic Text Summarization

We now review prior methods that have been used in automatic text summarization to estimate the quality of summarization systems. Since the goal of automatic summarization is to generate high-quality summaries, evaluation methods aim at estimating the difference between automatically generated summaries and reference summaries generated by humans. Following the introduced terminology, humans are functions f and automatic summarization are learned functions $\tilde{f}$. The task of an evaluation system is, as described previously, to learn a distance function $\tilde{d}$ which has to approximate the ”true” distance function d as good as possible.

We distinguish prior evaluation models whether they evaluate on a sentence-level or on a summary-level and discuss sentence-level evaluation in Section 7.2.1, manual summary-level evaluation in Section 7.2.2, and automatic summary-level evaluation in Section 7.2.3.

7.2.1 Automatic Sentence-Level Evaluation

In this section, we discuss sentence-level annotations which can be used to evaluate extractive summarization systems automatically.

Binary Classification

Early work in extractive summarization used information retrieval functions such as accuracy, precision, and recall as described above to evaluate the performance of summarization models (Edmundson, 1969; Goldstein, Kantrowitz, Mittal, & Carbonell, 1999; Jing, Barzilay, McKeown, & Elhadad, 1998). Edmundson (1969), for example, split texts into sets of individual sentences x_i and classified 25% of the sentences as "extract-worthy" leading to a sets of positive sentences p_i and a set of negative sentences n_i with sizes $|p_i| = \frac{1}{4}n$ and $|n_i| = \frac{3}{4}n$ where n equals to the size of $n = |X|$ (i.e., the number of sentences in x_i). The performance of summarization systems was assessed by estimating the labeling accuracy.

A problem observed with binary sentence classification is that human judges often disagree about which $n\%$ percent of the sentences are most relevant (Radev, Tam, & Erkan, 2003). Different sets of sentences can be similarly good. This observation has been named Summary Sentence Substitutability (Radev et al., 2003). Donaway, Drummey, and Mather (2000) show that the recall for the same automatically generated summary can vary from 0.25 to 0.75 depending on which human-created reference summary is used for comparison, which is problematic for an evaluation method.

Labeling the sentences according to their relevance can additionally be a great effort. However, given that all sentences have been labeled, evaluating the output of extractive summarization systems can be performed automatically without any human effort, which is a great advantage of sentence-level relevance annotations.

Precision and recall can be used instead of accuracy if the size of the sentences which have to be labeled as relevant is not predefined in advance.

Ordinally Scaled Relevance Labels

Instead of only using binary relevance labels such as *relevant* and *not relevant*, Relative Utility (RU) has been used to provide a more fine-grained assessment of individual sentence importance (Radev et al., 2003). With RU, annotators annotate each sentence individually with a utility score between 0 and 10 where a utility of 0 indicates irrelevant sentences and a utility of 10 indicates that a sentence is central to the topic. Since each sentence is judged individually, using a relative utility resolves the Summary Sentence Substitutability problem described above (Radev et al., 2003). A relative utility of a set of sentences is then assessed relative to the maximum possible score. Relative Utility with Subsumption additionally incorporates that sentences have a relative utility with respect to other sentences. Summarizers should, for example, not be rewarded if they include very similar sentences into the same summary. Radev et al. (2003) annotate sentence pairs whether they are subsumed and reduced the RU by a discount factor α in case of a subsumption.

7.2.2 Manual Summary-Level Evaluation

The big advantage of sentence-level annotations is that the set of sentences is already known before summarization models generate summaries. Even if it is an expensive endeavor, it allows annotating all sentences with relevance labels or relevance utilities as described above. Generated extractive summaries can be evaluated automatically as soon as all sentences are annotated. However, evaluation on a sentence-level is not appropriate anymore if summarizers generate summaries by modifying sentences, for example by deleting or adding words and phrases, since summarization is not a subset selection problem anymore and sentences which can be found in the summaries may not be present in the input. This observation means that it is not possible to annotate all sentences since the sentences which can potentially be contained in the summaries are not known before the summarization systems are applied. This issue is even more severe if abstractive summarizers are used since the output of abstractive summarizers might be even more different from the input than systems that only modify individual sentences.

One possible solution to this problem is to ask humans annotators to assess the quality of each created summary manually. Quality assessment of full summaries is not limited to content quality but can also be extended to linguistic quality. We, however, do not discuss linguistic quality in detail in this thesis (see Section 1.4).

Summary Evaluation Environment

To evaluate the content quality of summaries, the Summary Evaluation Environment (SEE) (C.-Y. Lin & Hovy, 2002b) has been used in the DUC shared tasks from 2001 to 2004 (Over et al., 2007). SEE was used by annotators to assess the content coverage of automatically generated summaries with respect to a reference summary. The annotators were able to choose if a generated summary covers 0%, 20%, 40%, 60%, 80%, or 100% percent of the content contained in a reference summary. SEE is therefore related to previously discussed functions such as Precision, Recall, and Accuracy since they estimate the amount of content overlap. Since the manual annotation was very time-consuming (Over et al., 2007), each automatically generated summary could only be judged against one single reference summary. This situation is problematic since it has been shown before that the results can depend strongly on the choice of the reference summary (Donaway et al., 2000).

The Pyramid Method

The Pyramid method (Nenkova, Passonneau, & McKeown, 2007) is another manual evaluation method and was used in the DUC 2005 shared task for the first time (Over et al., 2007). The key idea of the Pyramid method is to weight text snippets by their frequency with which they occur (semantically) in a set of reference summaries. To this end, text snippets have to be identified manually, similar to the information pieces we used in Section 5.3.3. Semantically similar text snippets are manually clustered into Summary Content Units (SCUs).

For each SCU it is manually counted in how many reference summaries the SCU appears. The quantity is used as weights for SCUs and indicates their importance. Formally, let SCU_i be an SCU and $y_1, \dots, y_n$ be n reference summaries. The weight w of SCU_i is defined by $w(SCU_i) = \sum_{j=1}^{n} \mathbb{1}_{y_j}(SCU_i)$ where $\mathbb{1}_{y_j}(SCU_i)$ indicates if SCU_i is contained in reference summary y_i. The name Pyramid method stems from the weighting scheme used for weighting the SCUs. SCUs with the maximum weight, which is usually a weight of 4, are located at the top of a pyramid, SCUs with a weight of 3 are located at the second-highest level, and so on.

The utility of a summary is determined by the sum of the weights of included SCUs whereby each SCU is considered only once even if it appears multiple times in a summary. Formally, let y_i be a summary, $w(SCU_j)$ the weight of SCU_j, and n the total number of SCUs. The utility u of y_i is defined by $u(y_i) = \sum_{j=i}^{n} \mathbb{1}_{y_j}(SCU_j)$.

Nenkova and Passonneau (2004) have already reported that a large-scale application of the Pyramid method is infeasible. Over et al. (2007) report a considerable effort for the annotation process in the DUC challenges. This additional annotation effort is unattractive for researchers, who prefer automatic methods such as ROUGE, which is validated by the few applications of the Pyramid method until today.

The need for more human annotation also introduces an additional source for annotation mistakes. Inspecting the annotations in the TAC 2008 dataset in detail reveals that this is not only a theoretical issue but has practical implications. We found several issues such as not annotating parent SCUs, missing SCUs in sentences, and different annotations for equal sentences in the annotations.

PEAK (Yang, Passonneau, & de Melo, 2016) is an attempt to automate the Pyramid evaluation (similar to work presented by Passonneau, Chen, Guo, and Perin (2013)). PEAK also requires reference summaries and is therefore not cheaper than automatic summary-level evaluation methods.

7.2.3 Automatic Summary-Level Evaluation

Human evaluation is costly and slows down the development of summarization systems since no fast evaluation of thousands of automatically generated summaries is possible. This fact motivates research to develop automatic evaluation models.

Evaluation without Additional Information

A simple and cheap evaluation is not to use any additional evaluation data. Model-free evaluation methods such as Jensen-Shannon divergence (Louis & Nenkova, 2013) do not require human input such as reference standard summaries and can be applied without any additional cost. The quality of these evaluation methods is, however, limited (see Section 7.4.3 for details), which makes applications problematic.

ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is the most popular used function for automatic evaluation in automatic summarization (Hong et al., 2014; Nenkova & McKeown, 2011; Yao, Wan, & Xiao, 2017) and was first used in the Document Understanding Conference (Over et al., 2007). ROUGE was used to evaluate the performance of many popular summarization systems (Erkan & Radev, 2004; Gillick et al., 2009; H. Lin & Bilmes, 2011; Mihalcea & Tarau, 2004). It is inspired by the BLEU evaluation method (Papineni, Roukos, Ward, & Zhu, 2002) and is based on measuring the lexical n-gram overlap of (stemmed) tokens between generated and gold standard summaries.

Formally, let $n - gram$ be a function which extracts all n-grams from a text. The ROUGE$_n$ score of a generated summary $\tilde{y}_i$ with respect to a reference summary y_i is defined as

$$\text{ROUGE}_n = \frac{\sum_{N \in s_n(y_i)} \min(\#_N(l_n(\tilde{y}_i)), \#_N(l_n(\tilde{y}_i)))}{|l_n(y_i)|} \tag{7.4}$$

where l_n and s_n are functions which return a list / a set containing all n-grams which occur in a text. Furthermore, let $\#_N(l_n)$ indicate the number of occurrences of n-gram N in list l_n.

The quality of ROUGE is often criticized in the research community. Sjöbergh (2007), for example, shows nicely how the ROUGE recall scoring can be fooled easily. A simple greedy language model based on the source documents extracts frequent bi-grams which are likely to occur in the reference summaries. The generated texts are merely lists of bi-grams and not meaningful sentences which cannot be considered to be summaries. However, they achieve superhuman ROUGE recall scores. In the TAC 2008 shared task (Dang & Owczarzak, 2008), both ROUGE$_2$ and ROUGE$_{SU4}$ score automatic systems higher than human summaries, which would lead to the conclusion that these systems are able to produce better summaries than humans. Furthermore, studies show that the correlation between ROUGE scores and human judgments may not be significant in non-newswire genres and other summary types (Feifan Liu & Liu, 2008). ROUGE also has many parameters (Graham, 2015), which makes reproduction and comparison of results problematic. Last but not least, ROUGE computes text similarity only based on simple string matching. Expressing the same information with different words is not rewarded by ROUGE.

AESOP Shared Task

The Automatically Evaluating Summaries Of Peers (AESOP) track in the Text Analysis Conferences (TAC) 2009-2011 aimed at developing systems that are able to automatically estimate the score of a summary for a given function. The AESOP tracks in TAC 2009 and TAC 2010 focused on two functions: a modified Pyramid score that measures summary content and the overall responsiveness score, which measures a combination of content and linguistic quality. Additionally, a third function that measures readability has been investigated in the TAC 2011 AESOP track. The systems in this shared task also considered reference summaries as additional information to evaluate a reference summary and are therefore as expensive as ROUGE in terms of required human annotation effort.

Other Works

Machine learning has been used to learn a linear combination of n-gram methods to evaluate summaries Giannakopoulos and Karkaletsis (2013). Mackie, Mccreadie, Macdonald, and Ounis (2014), Giannakopoulos (2013), and Cohan and Goharian (2016) investigate evaluation for microblog, multilingual, and scientific summarization, respectively. We focus on evaluating the information content of summaries and do not evaluate linguistic quality. This is, for example, captured by Pitler, Louis, and Nenkova (2004).

7.2.4 Discussion

Evaluation in automatic summarization is difficult due to the high complexity of the task itself and the huge amount of output possibilities. A manual evaluation should ideally be used to evaluate the output of summarization models, which is, however, too expensive. Automatic evaluation methods usually require reference summaries. The creation of reference summaries is, however, expensive and complex as well. Furthermore, the evaluation performance of automatic evaluation systems is much lower than usually assumed, which we will discuss in detail in the next section. After that, we present a new evaluation method which does not require reference summaries but only cheap pairwise preferences.

7.3 How to Evaluate an Evaluation Method

In this section, we discuss how evaluation models can be evaluated. Evaluating evaluation models is necessary since the evaluation models are only approximations of the true evaluation function. Therefore,

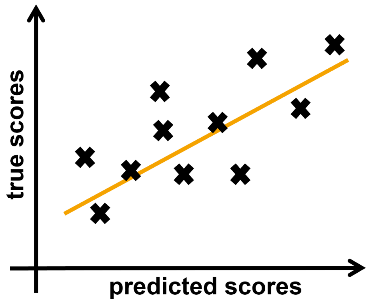


Figure 7.2.: Illustration of Pearson correlation coefficient

it is indispensable to first evaluate how well the evaluation methods work before they can be used for evaluation. We review in Section 7.3.1 a method used to evaluate the performance of the well-known ROUGE method. The idea is to compute the Pearson's correlation between scores predicted by an evaluation model and human annotators. We first argue why computing Pearson's correlation might not be a good choice in Section 7.3.2 and show in Section 7.3.3 that this method cannot be reliably used for evaluation. We discuss a more appropriate method, which does not suffer from the discussed disadvantages in Section 7.3.4. We use this method to evaluate a novel evaluation method based on pairwise preferences in Section 7.4.

7.3.1 Measuring Pearson's Correlation with Humans

One way to estimate the quality of an evaluation model is to compute the Pearson's correlation coefficient (also refereed to as Pearson's r or bivariate correlation) between human and automatically generated judgments. Let $\mathbf{\tilde{y}} = \tilde{y}_1, \dots, \tilde{y}_n$ and $\mathbf{y} = y_1, \dots, y_n$ be the predictions of an automatic and a human evaluation, respectively. For simplicity, we assume $\tilde{y}_i, y_i \in [0,1] \forall i \in 1, \dots, n$. We define $\dot{\tilde{y}} = \frac{1}{n} \cdot \sum_{i=1}^{n} \tilde{y}_i$ and $\dot{y} = \frac{1}{n} \cdot \sum_{i=1}^{n} y_i$ to be the respective sample means. Pearson's correlation coefficient is defined by

$$r(\mathbf{y}, \mathbf{\tilde{y}}) = \frac{\sum_{i=1}^{n} (y_i - \dot{y}) \cdot (\tilde{y}_i - \dot{\tilde{y}})}{\sqrt{\sum_{i=1}^{n} (y_i - \dot{y})^2} \cdot \sqrt{\sum_{i=1}^{n} (\tilde{y}_i - \dot{\tilde{y}})^2}}. \tag{7.5}$$

We provide an illustration of a linear correlation in Figure 7.2. In the example, the Pearson's correlation between true scores (i.e., scores provided by human annotators) and predicted scores is visualized by the orange line. Each data point can, for example, be a summary which has to be evaluated.

The Pearson's correlation coefficient ranges from scores of +1, which indicates a perfect correlation between $\mathbf{y}$ and $\mathbf{\tilde{y}}$, to -1 which indicates an inverse correlation. If Pearson's correlation equals to 0, we say that there is no correlation between $\mathbf{y}$ and $\mathbf{\tilde{y}}$.

The assumption which has been used in the past is that the quality of an evaluation model is high if the Pearson's correlation of model and human judgments is high. We show in Section 7.3.3 that this assumption is not true. Previously, we discuss more theoretical problems if Pearson's correlation is used.

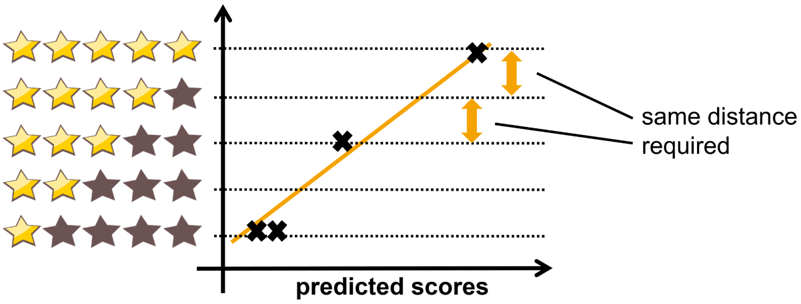


Figure 7.3.: Illustration of interval scaled values.

7.3.2 Problems with Pearson's Correlation

Pearson's correlation suffers from several drawbacks, which are discussed in the following.

Requires Interval Scaled Values

First of all, the application of Pearson's correlation requires interval scaled values. Interval scaling is also referred to as the third level of measurement after the nominal and ordinal scale as the first and second level of measurements. Interval scale requires in addition to the two requirements of ordinal scale (which are named and ordered values) that there is a proportionate interval between variables, which means that the distance between variables has to have a meaning and cannot change arbitrarily in different regions. Figure 7.3 illustrates that the distance between a 5-star rating and a 4-star rating has to be equal to the distance between a 4-star and a 3-star rating. Formally, it is required for every two variables a, b that $d(a, b) = d(a + c, b + c)$ holds where d is a distance measure.

Often used evaluation schemes such as a 5-point Likert scale (Likert, 1932) are, however, not proven to be interval scaled. A question in summarization is for example: "How well does the summary convey the key message of the input document?" which has to be answered by checking one of the following answer possibilities on a 5-point Likert scale: "very good", "good", "moderate", "poor", and "very poor". It is questionable that the distance between "very good" and "good" is the same as "good" and "moderate". Hence, Likert scales should only be considered to be ordinal scaled and not interval scaled.

Requires Linear Correlation

A second issue with Pearson's correlation is that it (only) measures the strength of the *linear* relationship between two judgments. Figure 7.4 shows two different evaluation scenarios. On the left, we see an evaluation system with a perfect linear correlation between true scores and predicted scores. On the right, we see a model with a perfect non-linear correlation between true scores and predicted scores. Pearson's correlation coefficient estimates the quality of the evaluation method on the left to be much better than the model on the right since only the left model has a linear correlation. It is, however, questionable why a perfect non-linear correlation should be assumed to be worse than a perfect or even

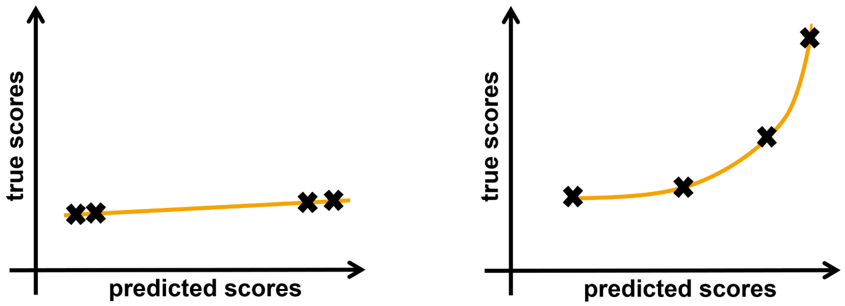


Figure 7.4.: Illustration of linear correlation on the left and non-linear correlation on the right.

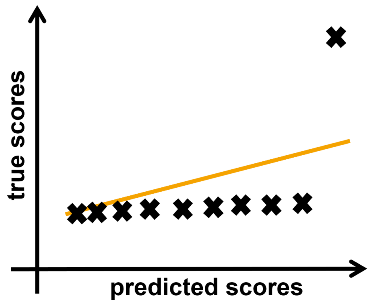


Figure 7.5.: Illustration of the sensitivity of Pearson's correlation to outliers

mediocre linear correlation, which holds in particular for the provided example in which the two best systems are rated to be much better (larger distance) than the two weaker systems even though the human judgments for all four systems are very close.

Sensible to Outliers

Pearson's correlation coefficient is also sensible to outliers. The model in Figure 7.5 has an almost perfect linear correlation for 9 out of 10 data points. However, the quality of the model is rated to be not very good since the model makes one "mistake" by predicting the score of the 10th data point to be similar to the other data points whereas the true score is much higher. Assigning a bad rating to this model is questionable since the data points are in perfect order.

Not Interpretable

The last criticism discussed in this section concerns the limited interpretability of Pearson's correlation coefficient. Due to the sensitivity to outliers and the estimation of linear correlation, it is difficult to estimate if an evaluation model is sufficiently good to be used in practice. When Pearson's scores are reported in studies, usually only qualitative judgments are made which do not allow a sound assessment of the consequences which have to be considered if the model is applied. It is, for example, hard to say what a Pearson's correlation of 0.70 means and if this correlation is good enough to be useful.

	System 1	System 2	System 3
Topic A	0.8	0.5	0.3
Topic B	0.4	0.3	0.4
Topic C	0.3	0.1	0.5
average	0.5	0.3	0.4

Table 7.1.: Illustration of scores from an imaginary summarization dataset. The performance of three different summarization systems 1, 2, and 3 are displayed for three different topics A, B, and C. Furthermore, we report the average performance for each system in the last line.

7.3.3 How to (not) Evaluate an Evaluation Method in Summarization

In this section, we discuss how evolution methods can be evaluated and how they should not be evaluated. We start with Pearson's correlation coefficient and consider rank correlation coefficients later as well.

Correlation Coefficients of System Averages

To compute Pearson's correlation coefficient, it is first necessary to select the scores which have to be compared. Table 7.1 provides an illustration of scores from an imaginary summarization dataset.

Since the goal of the evaluation process is to determine which summarization system is the overall best able to summarize, it seems to be reasonable to compute the average performance for each system across topics. This strategy was, for example, used to evaluate the ROUGE evaluation function in 2004 (C.-Y. Lin, 2004) and also more recently in a study, which analyzed the correlation of many different ROUGE versions (Graham, 2015). This strategy is, however, highly problematic since it reduces the number of scores for which the correlation is computed to a small number, namely to the number of participating systems. Only 17 summarization systems participated in the DUC 2004 summarization shared task.

ROUGE$_2$ achieves a Pearson's correlation of 0.695 if this evaluation strategy is used, which seems to be reasonably good. A closer look at the data, however, reveals that it is not difficult to achieve a high correlation in this setup. We list the averages of all scores assigned by humans in Table 7.2.

It can be observed that 15 of the systems have a score very close to 0.50 and the other remaining two systems have a score close to 0.27. The average and the variance of both groups are 0.50 ± 0.036 and 0.27 ± 0.016, respectively. The low standard deviation reveals that each group consists of similar values. Given such low standard deviations, every evaluation function which assigns arbitrary similar scores (e.g., $\tilde{y}^{\dagger}$) to all systems in the group of 15 and other arbitrary similar lower scores (e.g., $\tilde{y}^{*}$) to the systems in the group of 2 will have a high Pearson correlation with this sequence. Assigning, for example, a score of 1 to all systems in group $\dagger$ and a score of 0 to all systems in group $*$ results in a Pearson's correlation of 0.92. This correlation outperforms any ROUGE$_{.}$ score by far. We would be able to conclude that the newly generated poor 0/1 evaluation measure can be used as a replacement for human judgments if we utilized the argumentation used to establish ROUGE as a valid evaluation

System ID	Average scores across topics
2	$0.546^{\dagger}$
11	$0.503^{\dagger}$
19	$0.467^{\dagger}$
27	$0.469^{\dagger}$
34	$0.489^{\dagger}$
44	$0.543^{\dagger}$
55	$0.501^{\dagger}$
65	$0.548^{\dagger}$
81	$0.514^{\dagger}$
93	$0.497^{\dagger}$
102	$0.522^{\dagger}$
111	0.286^{*}
117	0.263^{*}
120	$0.546^{\dagger}$
123	$0.450^{\dagger}$
124	$0.511^{\dagger}$
138	$0.433^{\dagger}$

Table 7.2.: System results in the DUC 2004 shared task. We list the System ID and the average performance of each system across all topics. Two clusters of scores are indicated by * and †.

measure. We list the Pearson's correlation of ROUGE$_2$ and the proposed 0/1 evaluation measure along with two other versions of ROUGE in the line block of Table 7.3. It can be observed that ROUGE$_1$ and ROUGE$_{SU4}$ perform worse than ROUGE$_2$ and therefore, even worse than the poor 0/1 evaluation.

Instead of computing a correlation based on the actual values of the variable, rank correlation coefficients such as Spearman's and Kendall's rank correlation coefficient estimate the correlation based on the ranks of the variables (Kendall & Gibbons, 1990). Hence, only the order of the variables is relevant, whereas the actual scores are not relevant.

A result similar to the result for Pearson can be observed for Spearman's and Kendall's rank correlation coefficients in the first block of Table 7.3. The poor 0/1 metric achieves the highest Kendall correlation and the second-highest Spearman correlation, which would lead to the conclusion that the 0/1 measure is the best evaluation metric among the four tested measures.

We conclude that computing the average of systems scores across topics in a first step and computing correlation scores in a second step is problematic and cannot be reliably used to evaluate evaluation systems. Correlation scores are high due to the result of the average computation and not because the evaluation systems can predict summarization performance reliably.

Averaging Correlation Coefficients of Systems

Another way of aggregating the produced scores is to compute a correlation for each system individually in a first step and average the resulting correlation scores to obtain a single score. We present the resulting averaged correlations in the second block in Table 7.3. It can be observed that the correlation

1. step	2. step	R1	R2	SU4	0/1
avg. across topics	Pearson	0.445	0.698	0.559	**0.915**
	Spearman	0.355	**0.625**	0.434	0.559
	Kendall	0.235	0.426	0.309	**0.470**
system Pearson	average	0.286	0.260	0.262	-
system Spearman	average	0.240	0.234	0.206	-
system Kendall	average	0.165	0.163	0.142	-
overall Pearson	-	0.352	0.417	0.374	-
overall Spearman	-	0.326	0.393	0.339	-
overall Kendall	-	0.223	0.270	0.232	-
topic Pearson	average	0.355	0.483	0.416	-
topic Spearman	average	0.316	0.452	0.372	-
topic Kendall	average	0.231	0.336	0.276	-

Table 7.3.: Results for different correlation computation strategies in the DUC 2004 corpus. Columns "1. step" and "2. step" indicate which action is performed first and second, respectively. "system Pearson - average", for example, means that Pearson's correlation is computed for each system in a first step before the results are averaged in a second step.

scores are reasonably low, which indicates that none of the ROUGE versions is able to predict the scores well. This observation substantiates the previously drawn conclusion that the high correlation scores obtained with ROUGE are due to the averaging effect and not due to the predictive power of ROUGE.

We list the obtained averages of Pearson's correlation scores for ROUGE$_2$ for each system in Table 7.4. It can be observed that there is a correlation close to 0 for some of the systems (e.g., 102, 117, and 120). The highest correlations of 0.537 and 0.488 have been obtained for systems 2 and 19, respectively. This result shows that the variation of ROUGE's predictive power is quite high which means that the performances of different systems are estimated with a different accuracy, which is in addition to the low overall correlation problematic.

Correlation Coefficients without Averaging

A third way to evaluate the performance of an evaluation model is to compute the correlation coefficient of all values without any averaging. The vectors $\mathbf{y} = y_1, \ldots, y_n$ and $\tilde{\mathbf{y}} = \tilde{y}_1, \ldots, \tilde{y}_n$ contain the predictions for all systems in every topic. In the DUC 2004 shared task, $n = 17$ systems were applied in $m = 50$ topics which means that $|\mathbf{y}| = |\tilde{\mathbf{y}}| = n \cdot m = 17 \cdot 50 = 850$. Pearson's, Spearman's, and Kendall's correlations are displayed in the third block in Table 7.3. Again, we observe a much lower correlation than usually reported (Graham, 2015; C.-Y. Lin, 2004).

The problem with this approach is that scores from different topics are mixed even though the topics are independent. Summaries from different topics are never compared with each other. Only summaries within a topic are compared. Hence, it is not required to generate the same correlation in independent topics to create a good evaluation measure.

System ID	Pearson's correlation
2	0.537
11	0.288
19	0.488
27	0.382
34	0.247
44	0.296
55	0.214
65	0.243
81	0.212
93	0.336
102	0.040
111	0.128
117	0.079
120	0.071
123	0.301
124	0.310
138	0.337

Table 7.4.: Pearson's correlation of ROUGE$_2$ scores for each system in the DUC 2004 shared task.

We conclude that the best way to compute correlation coefficients for models for automatic summarization is to first compute correlation coefficients within each summarization topics and to compute the mean of the resulting correlation coefficients. The results for this analysis are listed in the last block of Table 7.3. Since Pearson's correlation suffers from various issues as discussed in Section 7.3.2, we propose to refrain from using Pearson's correlation and to use rank correlation coefficients instead. The last two rows of Table 7.3 are therefore the best candidates for a proper evaluation. The best performing system is ROUGE$_2$ with Spearman's and Pearson's rank correlation coefficients of 0.452 and 0.336, respectively.

In the next section, we propose to compute the agreement between humans and automatic evaluation systems, which is very similar to computing rank correlations. The proposed agreement score, however, does not consider summary pairs in which both summaries have the same score, which simplifies the evaluation since models only have to predict which summary is better instead of having a third option which states that both presented summaries have the same quality.

7.3.4 Estimating Evaluation Performance with Agreement

Let $\mathfrak{X} = X_1, \ldots, X_n$ be n summaries for a particular summarization topic and $\mathbf{R}$ a ranking over X such that $\mathbf{R}(X_i)$ indicates the rank of element X_i. The intuition of $\mathbf{R}$ is that a good summary is ranked highly (i.e., $\mathbf{R}(X_i)$ is small) whereas bad summaries are ranked at the end of the ranking (i.e., $\mathbf{R}(X_i)$ is big).

We define the agreement of an evaluation model with the ranking produced by human annotators by counting how often the evaluation model predicts the same preference label for two summaries with different ranks. Formally,

Donald Trump won the election and will become the 45th president.	$\succ$	*The U.S. Congress will certify the results on January 6, 2017.*

Figure 7.6.: Example of a pairwise preference annotation of two sentences. The first sentence is preferred over the second sentence because the first sentence contains more important information given that the information is not already known.

$$\text{agreement} = \frac{1}{m} \cdot \sum_{(X_i,X_j)\in X\times X, \mathbf{R}(X_i)\neq\mathbf{R}(X_j), i<j} \operatorname{sgn}(\operatorname{sgn}(\mathbf{R}(X_i)-\mathbf{R}(X_j)) \cdot \operatorname{sgn}(\tilde{\mathbf{R}}(X_i)-\tilde{\mathbf{R}}(X_j))+1) \tag{7.6}$$

where $m = |\{(X_i, x_j) \in X \times X : \mathbf{R}(X_i) \neq \mathbf{R}(X_j), i < j\}|$ the number of pairs with unequal ranks.

This evaluation of evaluation models is similar to the definition of *Agreement* and *Contradiction* from Owczarzak et al. (2012): *"Agreements occur when the two evaluation functions make the same distinction between System A and System B (...). Contradictions occur when both functions find a (...) difference between A and B, but in opposite directions."* A perfect evaluation model, which predicts the preference for all pairs correctly, yields an agreement of 1 whereas a random evaluation model, which always predicts the preference randomly, has an expected value of 0.5.

7.4 Context-free Evaluation with Pairwise Preferences

In this section, we present a novel evaluation model that estimates the quality of summaries based on pairwise preferences between sentences. Each pairwise preference of sentences indicates which sentence contains more important information. Figure 7.6 illustrates such a pairwise preference annotation for two sentences based on the example in Section 1. The information that Donald Trump won the election and will become the 45th president is considered to be more important than the information that the U.S. Congress will certify the results on January 6, 2017. Based on a set of pairwise preferences, the model estimates utilities for sentences in a first step and uses the sentence utilities in a second step to estimate utilities of summaries (Section 7.4.1).

One advantage of this approach is that sentence-level annotations can be used to evaluate summaries, which renders the complex and error-prone semantic comparison of long texts obsolete. Furthermore, pairwise preferences between sentences do not have to be consistent, which is an advantage since disagreements between different annotators (Gambhir & Gupta, 2017) do not require any additional treatment. Last but not least, the creation of high-quality reference is expensive and complicated, as discussed in Section 5.2.7. The presented model does not require reference summaries. Instead, pairwise preferences can be obtained cheaply.

The proposed usage of pairwise preferences between sentences is close to the idea of generating a ranking of sports teams by playing individual matches. Instead of competitions between teams, we observe competitions between sentences. The outcome of a match between teams equals to the annotation of a pair of sentences by a human annotator. Since different people can have different opinions about the importance of information (Gambhir & Gupta, 2017), we expect that one sentence will not always be preferred by humans similar to the situation that the better sports team does not always win against a weaker opponent. This fact is expressed by the winning probability between teams (or sentences).

7.4.1 Model

In this section, we present a model that estimates the utility of a summary based on preferences between the sentences which are contained in the summary. More generally, the model predicts the score of a set based on pairwise preferences of contained elements in the set.

Setup

Let $\mathfrak{X} = X_1, \ldots, X_n$ be n *sets* each of which containing *elements* from set $X = x_1, \ldots, x_m$. In the text summarization evaluation scenario, $X_1, \ldots, X_n$ are summaries and $x_1, \ldots, x_m$ are sentences. The formulated problem is, however, much more general. The sets X_i can, for example, also be sports teams or research groups and the x_j can be football players or researchers, respectively. Let $\mathbf{R}$ be a ranking function for the sets in $\mathfrak{X}$, i.e., $\mathbf{R}$ assigns a rank $\mathbf{R}(X_i)$ to every set in $\mathfrak{X}$. The model aims at learning $\tilde{\mathbf{R}}$. It can, unlike many applications of pairwise preference learning, not observe pairwise preferences between sets in $\mathfrak{X}$ directly but can only observe pairwise preferences between elements in X. This learning setup models the situation in which no pairwise preferences between elements in $\mathfrak{X}$ can be observed during training, which is, for example, the case when summarization corpora are newly created, and no summaries have been generated so far, but supplementary evaluation data z has to be generated for the corpus. Another situation in which the number of possible sets in $\mathfrak{X}$ is very large such that observing only a relatively small number of pairwise preferences between elements in $\mathfrak{X}$ is not sufficient to learn the ranking function $\tilde{\mathbf{R}}$. In such a situation, it would be appealing if a model could learn $\tilde{\mathbf{R}}$ based on observations between elements in X. The underlying assumption is that sets in $\mathfrak{X}$ are more likely to be ranked highly if they contain elements from X which have a high utility.

Estimating Sentence Importance

Let $\succ_X \subset X \times X$ a preference relation over elements in set X. We use $x_i \succ x_j$ as shorthand notation for $(x_i, x_i) \in \succ_X$. Let P be a list of k pairwise preferences. Based on P the model estimates utilities $u(x_i)$ of elements x_i such that

$$p(x_i \succ x_j) = \frac{u(x_i)}{u(x_i) + u(x_j)}, \tag{7.7}$$

where p is the probability that x_i is preferred over x_j according the preferences in P and $\sum_{l=1}^{m} x_l = 1$. Given that $X = \{x_1, x_2\}$ and $P = (x_1 \succ x_2), (x_2 \succ x_1), (x_1 \succ x_2), (x_1 \succ x_2)$, the model assigns utilities of $p(x_1) = 0.75$ and $p(x_2) = 0.25$ since the probability that x_1 wins against x_2 is 0.75.

The model in Equation 7.7 is called Bradley-Terry model (Bradley & Terry, 1952) and is well-known in sports where it can be used to generate *power rankings* for sports which only have win/loss outcomes. A power ranking is used to describe a ranking that does not only rank the individual teams but also assigns a utility to each team, which is also called the *skill* of the team.

Zermelo (1929) has already proposed an algorithm to find a maximum-likelihood estimator for the Bradley-Terry model. Iteratively applying Equation 7.8 for all elements $x_i \in X$ until the difference between two iterations is sufficiently small leads to a unique maximum if the utilities are normalized after each iteration.

$$u(x_i) \leftarrow \text{wins}(x_i) \sum_{j=1, j\neq i}^{k} \frac{\#(x_i, x_j)}{u(x_i) + u(x_j)} \tag{7.8}$$

In Equation 7.8, $\text{wins}(x_i)$ denotes the number of times that x_i was preferred over another element (i.e., $\text{wins}(x_i)$ equals to the number of times with $(x_i \succ x_l) \in p$ for $l = 1, \ldots k, i \neq l$) and $\#(x_i, x_j)$ the number of preference between elements x_i and x_j.

Normalization is required to get a unique solution since every multiple of a solution is also a correct solution. We initialize all utilities $u(x_i)$ equally by setting $u(x_i) \leftarrow \frac{1}{m}$ where $m = |X|$ as described above.

Estimating Summary Quality

Based on the utilities for individual elements, we define the utility of a set X_i as

$$\dot{u}(X_i) = \sum_{j=1}^{|X_i|} u(x_j). \tag{7.9}$$

The score of a set is therefore defined as the sum of the utilities of all contained elements. Defining the utility of a set as the sum of it's elements is a simplifying assumption which is often made (Herbrich, Minka, & Graepel, 2006; T.-K. Huang, Lin, & Weng, 2006) but not always correct in practice (Ramond, 2010). We therefore make some modifications to Equation 7.9 to make it more appropriate for an application in text summarization.

We first define a weight w_i for each sentence x_i according to

$$w_{i,j} = \frac{||x_j||}{||X_i||}, \tag{7.10}$$

where $||.||$ denotes the length of summary X_i and sentence x_i measured in number of characters. The intuition of the weight is that a sentence contributes more to the overall score of a summary if it is longer.

Hence, the score of a summary will decrease if a large fraction of the summary is occupied with a poor sentence. We update Equation 7.9 by incorporating the sentence weight which yields

$$\dot{u}(X_i) = \sum_{j=1}^{|X_i|} w_{i,j} \cdot u(x_j). \tag{7.11}$$

Sentences in summarization dataset often contain redundant information. The utility in Equation 7.11, however, does not consider redundancy. Including a sentence x_j twice would, therefore, result in adding the utility of x_j twice to the summary utility. Since this behavior is not desirable for the evaluation measure, we include a redundancy penalization that does not reward redundant information. For a summary X_i, we reduce the score of sentence x_j by

$$\widetilde{u(x_j)} = u(x_j) \cdot \frac{1}{||x_j||} \cdot \sum_{g \in x_j} \frac{\#(g, x_j)}{\#(g, X_i)} \tag{7.12}$$

where $\#(g, x_j)$ and $\#(g, X_i)$ denote the number of occurrences of the bigram g in x_j and X_i, respectively. $||x_j||$ denotes the length x_j. Equation 7.11 is modified accordingly:

$$\dot{u}(X_i) = \sum_{j=1}^{|X_i|} w_{i,j} \cdot \widetilde{u(x_j)}. \tag{7.13}$$

The model is now able to weight sentences according to their length and to consider redundancy. However, the model is only able to compute the utility for summaries which only contain known sentences, i.e., sentences x_j for which preferences have been collected and the utility $u(x_j)$ is known. Summarization systems are, however, free to create new sentences which cannot be known before the summarization system has been applied to a dataset, which means that the evaluation model has to be able to estimate the utility of summaries Y_i which do not (solely) contain sentences from set X but also for new, unseen sentences which are not known at the time when the dataset is created. Let Y be the set of unseen sentences. Since we cannot collect pairwise preferences for sentences that are not known yet, we modify the utility computation one last time. Instead of summing the utilities of the sentences which are contained in a summary Y_i, we sum the utilities of the sentences in X (where X is a set containing all sentences in the input documents as defined above) which are most similar to the sentences in Y_i. Formally,

$$\dot{u}(Y_i) = \sum_{j=1}^{|Y_i|} w_{i,j} \cdot u(\arg\max_{x \in X} \text{sim}(x, y_j)). \tag{7.14}$$

As similarity measure sim, we use the average of a Cosine similarity based on TF-IDF vectors and the Jaccard similarity.

The Cosine similarity of two vectors $\mathbf{a}, \mathbf{b}$ with length n is defined as

$$\text{sim}_{cos} = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}|_2 \cdot |\mathbf{a}|_2} \tag{7.15}$$

where $\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i \cdot b_i$ is the dot product between two vectors and $|\mathbf{a}|_2$ and $|\mathbf{b}|_2$ are the L^2 norms of $\mathbf{a}$ and $\mathbf{b}$, respectively. The L^2 norm $|\mathbf{a}|_2$ for a vector a with length n is defined as $|\mathbf{a}|_2 = \sqrt{\sum_{i=1}^{n} a_i^2}$. We use TF-IDF vectors to put a stronger weight on words which occur less frequently in a background corpus. The idea is that frequently occurring words are not substantially indicative of the content and therefore not important for similarity computation. Examples for English words which occur frequently regardless of the topics are words such as "he/she", "the", "a", etc.

The Jaccard similarity for two sets A and B is defined as

$$\text{sim}_{jac} = \frac{A \cap B}{A \cup B} = \frac{A \cap B}{|A| + |B| - |A \cap B|} \tag{7.16}$$

Hence, the Jaccard similarity measure the word overlap of two sentences without considering any word-level weighting. It can therefore be used to estimate syntactic similarity.

Hence, the combination of both Cosine and Jaccard similarity measures allows to both rely on the similarity computation on lexical similarity (Jaccard similarity) and with a focus on important content words (Cosine similarity). We therefore set

$$\text{sim} = \frac{1}{2} \cdot \text{sim}_{cos} + \frac{1}{2} \cdot \text{sim}_{jac}. \tag{7.17}$$

7.4.2 Obtaining Pairwise Preferences

We discuss in the following two sections, three different ways to obtain pairwise preferences that can be used to train the previously described model. Section 7.4.2 discusses how pairwise preferences can be generated with the help of human annotators and Section 7.4.2 shows how already available evaluation data in the form of reference summaries or Pyramid annotations can be used automatically generate pairwise preferences, which can, in turn, be used to train the presented model.

Human Annotations

The first way to obtain pairwise preference annotations is to select two sentences from a set of sentences, present the sentences to a human annotator, and to ask the human annotator which of the two sentences contains more important information.

Let X be the set of all sentences which are contained in the source documents in a given summarization topic. A simple method to sample two sentences from set X is a random sampling strategy which samples pairs of two different sentences with probability $p((x_i, x_j)) = \frac{1}{|X|^2} \forall (x_i, x_j) \in X \times X, i \neq j$ and $p((x_i, x_j)) = 0 \forall (x_i, x_j) \in X \times X, i = j$.

For each sampled sentence pair (x_i, x_j), we ask an annotator to label the pair with a pairwise preference such that $x_i \succ x_j$ or $x_j \succ x_i$. The pairwise preference label indicates whether x_i or x_j contains more important information. The pairwise preferences are used to build set P, which is in turn used to train the model described in Section 7.4.1.

To reduce the number of required human annotation effort, we apply a *smooth propagation of knowledge*. The idea is that we do not only obtain information about the sampled sentence pair but also about pairs that are similar to the sampled pair. To estimate how much information can be transferred from one to another pair, we rely on estimating the similarity between pairs of sentences. As similarity function between individual sentences, we use the same mixture of Jaccard and Cosine similarity as described above. We define the similarity of the pair (x_{i_1}, x_{j_1}) and the pair (x_{i_2}, x_{i_2}) to be

$$\text{sim}((x_{i_1}, x_{j_1}), (x_{i_2}, x_{j_2})) = \text{sim}(x_{i_1}, x_{i_2}) \cdot \text{sim}(x_{j_1}, x_{j_2}) \tag{7.18}$$

Given a human preference for (x_{i_1}, x_{j_1}), we transfer $\text{sim}((x_{i_1}, x_{j_1}), (x_{i_2}, x_{j_2}))$ of the gained knowledge to pair (x_{i_2}, x_{j_2}). Transferring information means in this context that we generate additional preferences labels based on human preference labels. If a human annotated that he/she prefers x_{i_1} over x_{j_1}, i.e., $x_{i_1} \succ x_{j_1}$, we add additional partial preference labels to the set P of annotated preference labels. The weight of additionally created preference labels equals to the previously described similarity.

Automatically Generated Preferences

We present in the following two methods to generated pairwise preferences automatically instead of asking humans to generate preferences manually, which is possible because additional evaluation data is already available for some corpora such as the DUC and TAC summarization datasets. The key idea is to reuse references summaries or Pyramid annotations to create pairwise preferences automatically.

Preferences can be generated based on the similarity of sentences in the input texts and in the reference summaries. Let X be the set containing all sentences in the input documents and Z be the set of sentences in the reference summaries. We define a utility for a sentence $x_i \in X$ with respect to the reference summary sentences Z according to

$$u(x_i) = \max_{z \in Z}(\text{sim}(x_i, z)) \tag{7.19}$$

According to the utility, x_i will receive a high score if a similar sentence appears in a reference summary. If no similar sentence is in the gold standard summaries, the sentence will receive a rather low score.

Let $(x_i, x_j) \in X \times X$ be a sampled sentence pair (e.g., sampled with a random sampling strategy as described above). Based on the automatically estimated utility in Equation 7.19, we generate the preference label $x_i \succ x_j$ if $u(x_i) > u(x_j)$ and the preference label $x_j \succ x_i$ if $u(x_j) > u(x_i)$. The preference label can be generated randomly in the rare case of $u(x_i) = u(x_j)$.

Preferences can also be generated based on Pyramid scores. Given that Pyramid annotations are available (as in the TAC 2009 corpus, for example), we can define the utility of sentences to be equal to the sum of the weights of the matched SCUs (similar to the Pyramid method). Annotations are, however, only available for sentences in the reference summaries (i.e., in Z) and not for sentences in the input documents (i.e., in X). Hence, we reuse the idea from above and search for sentence z in Z that is most similar to sentence $x_i \in X$ and use the utility of z instead. Formally,

$$z = \arg\max_{z_j \in Z} \text{sim}(x_i, z_j) \tag{7.20}$$

and set the score of x_i to

$$u(x_i) = \sum_{SCU_j \in z} w(SCU_j) \tag{7.21}$$

where $SCU \in z$ are all unique SCUs contained in z and $w(SCU_j)$ denotes the weight of the j-th SCU as defined by Nenkova and Passonneau (2004) (see Section 7.2.2 for details).

7.4.3 Evaluation

We provide in this section a detailed analysis of the proposed evaluation method based on pairwise preferences.

For the experiment, we use eight topics from two popular multi-document summarization datasets, the DUC 2004 and TAC 2009 corpora (see Section 5.2 for more detail about the datasets). Each topic in the datasets contains ten source documents. Each topic contains automatically generated summaries which were produced by automatic summarization systems in the DUC 2004 and TAC 2009 shared tasks. Humans evaluated all automatically generated summaries. Each summary was labeled with a score from 1 to 5 (DUC 2004) and 1 to 10 (TAC 2009), which indicates the quality of the information content of the summary. Evaluation of grammatically, writing style, and other relevant metrics is not included in this score.

In the following, we report the agreement as described in Equation 7.6 for various experiments. Let $(Y_i, Y_j), i \neq j$ be two automatically generated summaries. The pairwise preference prediction $Y_i \succ Y_j$ of an evaluation model agrees with the human judgment if the human score for Y_i is bigger than the human score for Y_j. As discussed in Section 7.3.4 we do not consider ties. We use the abbreviations **JS** (Jensen-Shannon), **R1** - R4 (ROUGE-1 - ROUGE-4), **SU4** (ROUGE-SU4), and **PY** (Pyramid (Nenkova &

	JS	R1	R2	R3	R4	SU4	PL_{man}
DUC 2004	0.480	0.651	0.639	0.649	0.606	0.558	**0.673**
TAC 2009	0.565	0.638	0.668	0.660	0.674	0.663	**0.688**

Table 7.5.: Agreement of preference-based evaluation as defined in Equation 7.6 of different versions of Jensen-Shannon, ROUGE and our novel model based on manually labeled pairwise preferences.

Passonneau, 2004)) to denote the evaluation models with are used as reference systems. The reference evaluation models are discussed in detail in Section 7.2.

Learning to Evaluate with Human Preferences

In the first experiment, we investigate how well the proposed model performed if human preferences are used as evaluation data (i.e., as set P). As input, we use 200 pairwise preference annotations for each topic. The sentences have been sampled randomly, and we used the previously described smoothed knowledge transfer. The results are shown in Table 7.5.

Column **PL** denotes the performance of the presented preference-learning based model. On average, our model achieves an agreement of 0.673 in DUC 2004 and 0.688 in TAC 2009. This result means that 67.3 and 68.8 percent of all pairs of manually rated summaries were predicted correctly, respectively. The new model outperforms the best versions of ROUGE in the respective corpora (SU4 with 65.1 percent in DUC 2004 and R2 with 66.0 percent in TAC 2009).

The preference-based model needs much less annotation effort than ROUGE with an average annotation time of 53 and 54 minutes per topic for the DUC 2004 and the TAC 2009 dataset, respectively. This information is in particular important since one goal is to develop a cheap evaluation framework.

Learning to Evaluate with Human and Automatic Preferences

In the next experiment, we investigate whether additional automatic annotations based on already available reference summaries and Pyramid annotations can further improve model performance. To this end, we generate 200 additional pairwise preference annotations based on reference summaries and/or Pyramid annotations in addition to the 200 manual annotations per topic. Table 7.6, column **man+ref** contains the results for 200 manual + 200 automatic reference summary-based annotations; column **man+py** contains the results for 200 manual + 200 simulated Pyramid score-based annotations; and column **man+ref+py** contains results for 200 manual + 200 reference summary-based + 200 Pyramid score-based annotations. The results show that the agreement improves with additional simulated annotations based on reference summaries in DUC 2004 by 5 percentage points. Additional annotations increased the agreement in TAC 2009 by 3 percent points, which leads to the conclusion that already available reference summaries and Pyramid annotations can indeed be used in order to substitute more human preference annotations, which makes the trade-off between performance and annotations effort of the preference-learning model even better.

	JS	R1	R2	R4	SU4	$\text{PL}_{\text{man}\cup\text{ref}}$	$\text{PL}_{\text{man}\cup\text{py}}$	$\text{PL}_{\text{man}\cup\text{ref}\cup\text{py}}$
DUC 2004	0.480	0.651	0.639	0.606	0.558	**0.722**	n/a	n/a
TAC 2009	0.565	0.638	0.668	0.674	0.663	0.682	0.707	**0.717**

Table 7.6.: Agreement of different versions of ROUGE and the new models based on human and automatically generated pairwise preferences in addition to manually labeled preferences.

	R1	R2	R4	SU4	PL_{ref}	PL_{py}
DUC04	0.651	0.639	0.606	0.558	**0.716**	n/a
TAC09	0.638	0.668	0.674	0.663	0.644	**0.709**

Table 7.7.: Agreement with human judgments for reference systems and our model fed with only automatically generated preferences labels.

Learning to Evaluate Solely Using Automatic Preferences

Now, we investigate whether using automatic preferences is already sufficient to produce reasonable good results. This question is interesting since many old summarization datasets contain reference summaries or even Pyramid annotations. The pairwise preference model can be applied without creating any human annotation (i.e., without any additional human effort) if the model performs well with automatic preferences. Table 7.7, columns **ref** and **py** contain the results of an experiment where we sampled 1,000 pairwise annotations automatically. Without any additional annotation effort, the new model is able to perform much better than ROUGE at DUC 2004. In TAC 2009, the model achieves similar performance as the best performing evaluation based on Pyramid annotations. The model does, however, not improve beyond ROUGE if preferences are only generated based on reference summaries without using preference generated based on Pyramid annotations.

Convergence

In the next experiment, we investigate how the agreement changes over time with an increasing amount of annotations. Figure 7.7 shows how agreement improves with more annotations. We sampled n annotations (horizontal axis) randomly from the set of human annotations and averaged the resulting agreement scores (vertical axis) of 100 runs to obtain a reliable average. We observe a continuous improvement of agreement in four topics in the TAC 2009 dataset. Hence, generating more annotations may further improve the performance of the evaluation system. We see also that the agreement differs in the different topics indicating that the model performs better in some topics and worse in others.

Ranking Evaluation

We now investigate the ranking generated by our model directly. Since individual sentences are annotated in the TAC 2009 corpus with SCUs, we can generate a ranking of the sentences and directly compare the ranking with the ranking generated by our model. Table 7.8 shows the percentage of correctly ordered sentence pairs (similar to Kendall's τ) for the new model with and without smoothed sampling.

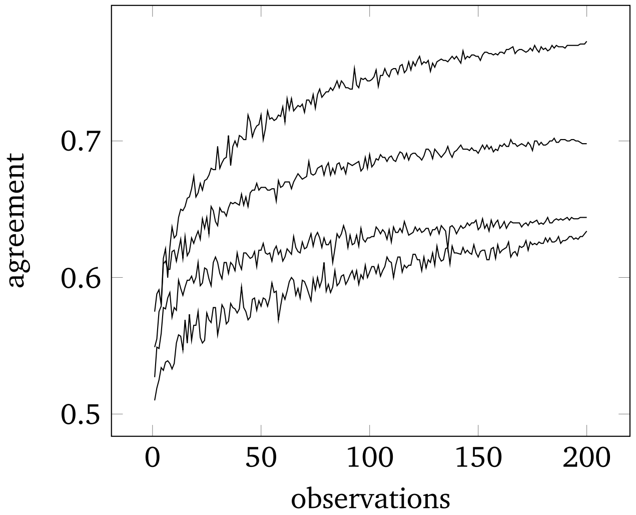


Figure 7.7.: Agreement trajectories averaged over 100 runs per topic in the TAC 2009 corpus. Each curve illustrates the learning in one particular topic.

non-smoothed			smoothed		
PL_{man}	PL_{py}	PL_{ref}	PL_{man}	PL_{py}	PL_{ref}
0.683	0.987	0.661	0.727	0.941	0.698

Table 7.8.: Percentage of correctly ordered sentence pairs in the TAC 2009 corpus for both a non-smoothed and a smoothed sampling.

Smoothed sampling improves the ranking of the model if 200 manual or 200 reference summary-based preferences are used in the TAC 2009 corpus. Given that we can sample pairs based on Pyramid scores, the model is able to reconstruct the ranking almost perfectly if we do not use smoothed sampling. With smoothed sampling, the performance decreases in this case. The result confirms the previously observed performance at summary scoring where preferences based on Pyramid annotations performed best, followed by manually generated preference annotations.

7.4.4 Conclusions & Outlook

Evaluating automatically generated summaries is a challenging task, and creating evaluation data that are required by evaluation methods such as ROUGE or Pyramid is laborious and expensive. We proposed an alternative model that does not rely on reference summaries or Pyramid annotations but only on simple pairwise preferences between sentences.

We showed in our experiments that the proposed model is able to perform better than the current state-of-the-art ROUGE method with less expensive annotations and that humans are able to provide useful feedback in the form of pairwise preferences. In combination with already available reference summaries and Pyramid annotations, we were able to simulate more annotations, which improved performance further.

We conclude that gold standard summaries are not the only valuable human feedback, which can be used for summary evaluation. Investigating other kinds of feedback, such as pairwise preferences might be a promising future research direction.

It would be interesting to investigate whether the number of required preferences can be reduced with smarter sampling methods without losing prediction accuracy. Active learning methods can, for example, be used to replace the simple random sampling strategy which we used in the experiments. Additionally, the investigation of more sophisticated similarity functions can potentially improve the model's performance.

The converges experiment showed that the performance of the newly presented model differs across topics. One reason for the differences might be the size of the topics. Topics with many sentences in the input documents might require more samples to be covered reasonably well. It would be interesting to investigate why the performance of the model varies in different topics to be able to improve the prediction agreement further.

7.5 Summary

In this chapter, we discussed the development of automatic evaluation methods. Developing automatic methods is important for summarization and many other tasks such as machine translation in which no obvious ways are available to evaluate systems automatically.

Manual evaluation such as SEE and the Pyramid method are expensive and complicated. Hence, they cannot be applied on a large scale or in a short amount of time. Both is necessary if automatically generated summaries have to be evaluated during the development of summarization models as well as for final evaluation.

Available automatic methods usually use reference summaries as evaluation data, which is problematic for two reasons. First, creating reference summaries is expensive. Second, estimating the semantic similarity of two texts is a challenging and yet unsolved problem that renders automatic methods error-prone.

We presented a new evaluation method which does not require reference summaries but only cheap pairwise preferences. We have been able to create sufficient evaluation data with less than one hour of effort per topic to perform better than the most frequently used versions of ROUGE.

We also showed that computing the Pearson's correlation coefficient for averages of system scores is not a reliable method to evaluate evaluation methods. Instead, we propose to compute rank correlations within topics or the pairwise preference agreement between human judgments and automatic evaluation systems.

In general, we believe that much more effort has to be invested in making evaluation in summarization more reliable since good science cannot be performed without reliable evaluation of experiments.

Research for better evaluation methods is not only crucial for extractive summarization but also for abstractive summarization in which the diversity of automatically produced summaries is even higher than in extractive summarization.

Part III

Wrap-up

In the last part, we wrap-up this thesis by revisiting the research questions, discussing limitations of this work, and provide some final remarks.

In Chapter 8, we revisit each research question and describe how this thesis contributes to answer the questions. We summarize the main conclusions of this work. Furthermore, we discuss potential future work to advance context-free information importance estimation further. As claimed in this thesis, advancing context-free information importance estimation is necessary to develop reliable and robust importance estimators in the future.

We have already noted some very important problems in the introduction that are beyond the scope of this thesis, but are highly relevant for developing reliable and useful systems that can assist humans to get efficient access to most important information. In Chapter 9, we discuss limitations that have to be addressed on the quest towards context-free importance estimation.

We close this thesis in Chapter 10 with some final remarks.

8 Conclusions and Future Work

In this chapter, we conclude this thesis which has been guided by six research questions related to information importance estimation. For each question, we discuss which conclusions can be drawn and discuss potential future work.

8.1 How can information importance be defined in a meaningful way?

In Chapter 2, we aimed at finding answers to the question of how information importance can be defined. We discussed that prior work is mostly concerned with estimating the amount and usefulness of data. More recently, researchers focused on the differentiation between data and information and tried to find frameworks with which the amount of semantic information can be estimated. We concluded that prior discussions are related to the question of how information importance can be defined. However, we worked out why prior work does not provide answers to the question by providing counter-examples, which showed that information importance is rather independent of data and information amount. Hence, we discovered a research gap that we aimed to close by providing a definition of information importance. Having a more formal definition of importance is important to avoid relying on slippery concepts in people's minds. Instead, we aim for concepts to be as explicit as possible to be able to discuss properties precisely, analyze the pros and cons, and infer implications of the definitions.

In the first definition (see Definition 7), we defined information importance to be equal with the change it causes in the behavior of a person. The underlying intuition is that information should be considered to be important if it changes the conduct of a person and that information should be considered to be rather unimportant if it does not have a big impact on the behavior of the audience. We concluded, however, that this behavior-focused definition does not meet the anticipated characteristics of information importance and revised the definition. In the second definition of information importance (Definition 8), we equated information importance with the change it imposes on to the course of life of the recipient. We concluded that the second definition matches the intuition of information importance better than the first definition and analyzed its implications. We concluded that information importance as defined in Definition 8 is personalized, time-dependent, and non-additive.

Furthermore, the provided definition defines information importance without using frequently used signals in automatic information importance estimators such as information position and information frequency. Due to the lack of causal relationship, these features cannot be used to detect information importance reliably. This conclusion motivates the research for context-free information importance estimators.

We do not expect that the provided definition of information is the only possible or reasonable definition and invite other researchers to contribute other definitions with different properties. As soon as multiple definitions are available, similarities and, more importantly, differences between definitions can be discussed to find a good definition in the end which matches the meaning of the abstract concept of information importance best.

8.2 How can information importance estimation abilites be assessed?

After defining information importance, we discussed how information importance estimation capabilities can be tested in Chapter 3. Most prominently in the scientific literature is the task of automatic summarization, which aims at condensing the most important information from texts into a short summary. Based on the definition of importance from Chapter 2, we were able to provide an intensional definition of optimal summaries for groups consisting of one or more people. In summarization datasets, optimal summaries are usually specified by example, which has severe implications for the field of summarization. Summaries are not evaluated by checking if desired properties such as linguistic quality or focus are met, but by computing a syntactic similarity to the provided reference examples. The optimal summaries according to Definition 17 are the summaries that contain the most important information with respect to the information consumers. A key conclusion is that containing the most important information pieces and being representative of the input are two potentially conflicting properties in summarization. Representative summaries do not necessarily contain the most important information, and summaries that contain the most important information are not necessarily representative of the input. This fact is in particular important when research in automatic summarization moves away from newswire summarization, in which both representative and importance are correlated, to other domains and to more heterogeneous summarization setups in which both are not correlated anymore.

Furthermore, we concluded that the task of automatic summarization is suboptimal to test information importance abilities and proposed three different new tasks based on utility prediction, pairwise preference prediction, and ranking. With the new tasks, assessing information importance abilities can be performed without being concerned with linguistic quality, redundancy, or arbitrary length restrictions. Instead, the new tasks put more emphasis on information important estimation, which allows a more focused and better-structured investigation of automatic importance estimators.

8.3 Why do we need context-free information importance estimation?

Chapter 4 focuses on the question why we need context-free information importance estimators. Before we motivated context-free information importance estimation, we first clarified what we exactly mean with the term ’context’. To this end, we defined the terms semantic context and syntactic context. In both cases, context refers to the information or the text which surrounds the information nugget whose importance has to be estimated. The key idea of most summarization systems is to analyze the document in which information appears to estimate the importance of information. This strategy is very successful in newswire summarization. We analyzed why this is the case. We concluded that the writing style of journalists results in the fact that sentences at the beginning of newswire documents are likely to contain a lot of important information. Furthermore, we viewed journalists as an ensemble of strong importance estimators. Journalists report more important information more frequently, which results in the situation that more important information is often also more frequently included in newswire summarization topics than less important information.

We have motivated context-free summarizers already in the introduction of this thesis, where we used three information nuggets belonging to the U.S. presidential election in 2016. In this example, all three information nuggets have the same frequency since every nugget only appears once. Furthermore, no ordering of the information nuggets is provided such that the position of the information nuggets cannot be used for importance estimation. As a second example, we used a soccer transcript. In a soccer match, most important events such as goals and red cards can occur at any time and are usually rather infrequent. Hence, a position feature cannot be used to summarize soccer transcripts reliably. Extracting all infrequent information (basically the inverse strategy to newswire summarization) is also not reliable. In a match with only one yellow card, for example, it is not good to consider this information as important.

We then analyzed many prior works in automatic summarization with respect to the importance signals they use. We found that most summarizers rely on contextual importance signals. This observation is reasonable since these summarizers have been developed with newswire summarization in mind. We are, however, concerned with more heterogeneous summarization scenarios in this thesis in which we do not assume that easy-to-exploit signals for importance estimation can be extracted from the input documents. This focus motivates the development of context-free summarization methods and hence answers the research question of this chapter.

8.4 How can challenging datasets for information importance estimation be created cheaply?

The overall aim of this thesis is to advance the development of automatic information importance estimators. We focus on context-free information importance estimation because we believe that a reliable estimation of information importance is not possible by relying on document-derived importance signals as analyzed in the previous chapter.

Document-derived importance signals are, however, very good importance indicators in newswire summarization, which renders context-free information importance useless. Hence, we need new and more challenging datasets to show the limitations of currently available systems. Furthermore, we need large datasets to train summarization systems based on machine learning. We concluded that available summarization datasets are small and/or focused on the newswire genre.

To mitigate the problem of a lack of training and evaluation data, we proposed in Chapter 5 new approaches to cheaply generate challenging heterogeneous summarization datasets. The key idea is to reverse the traditional corpus construction approach: instead of asking humans to select input documents and to write reference summaries, we proposed to find already available summaries and to search for appropriate input documents. The input documents are supposed to contain the information contained in the summary as well as unimportant and/or unrelated information. We generated two new corpora with the new corpus construction approach: *h*MDS and auto-*h*MDS. *h*MDS is larger than traditional multi-document summarization corpora, and the creation was relatively cheap. However, its creation still required human effort and does therefore not scale to the generation of very large summarization corpora. To increase the size and to reduce the cost per topic further, we modified the corpus construction approach and entirely automatically created auto-*h*MDS, a large, heterogeneous, multi-lingual, multi-document summarization corpus.

Similar to research in question answering, in which it has been shown that the problems represented in datasets can be solved with simple clues, we believe that today's famous newswire summarization datasets will also be criticized for containing simple clues such as position and frequency signals which do not require any language understanding. Hence, generating more difficult summarization datasets that require language understanding to produce good summaries is crucial to advance automatic summarization further.

In future work, our corpus construction approach can be used to create even larger summarization datasets. *h*MDS and auto-*h*MDS only use featured Wikipedia articles due to the high quality of the summaries. However, it may also be reasonable to consider all Wikipedia articles as summaries to create a much larger version of auto-*h*MDS. Furthermore, we only generated data in English and German. Wikipedia articles are, however, also available in many other languages which can be used for dataset generation as well. Furthermore, not only Wikipedia should be considered as a source for summaries. The proposed approach also works with other sources as long as the validity criteria described in the approach are met.

In the previous section, we already discussed the newly presented tasks for information importance estimation. The big advantage of the approaches is that data for the new tasks can be produced automatically based on already available summarization dataset. In Section 6.3.5 and Section 6.3.5, we showed in experiments how summarization dataset can be used for automatic dataset created. It would be rather simple to convert available datasets into datasets for ranking, pairwise preference prediction and/or utility prediction. More dataset for the newly proposed tasks can also be manually generated. Since the generation of data for the tasks is much simpler than writing summaries, crowdworking platforms might be suitable for creating large datasets. Our initial efforts showed, however, that additional quality control might be necessary to produce high-quality datasets.

8.5 How can machines learn context-free estimation of information importance?

A central question of this thesis is how machines can learn to estimate information importance without considering the context in which the information appears. We believe that context-free information importance estimation is an essential capability of truly intelligent automatic summarization systems. Hence, we developed a fully context-free summarization system in Chapter 6. The key idea of context-free information importance estimators is to acquire world knowledge in a first learning step, which does not have to be targeted to summarization and to apply the learned world knowledge to summarization in a second step.

For example, the fact that the information who won a U.S. presidential election is important information can be learned from many sources. We opt for learning such world knowledge from summarization datasets by learning which information is usually promoted from the source document to the summary. Other options would be to consult a database which enumerates the most powerful people in the world. Since the U.S. president will be among these people, it could also be inferred from this database that a change of the position of U.S. president is an important event that should be reported to affected people.

The presented summarization model is able to estimate information importance without deriving any importance signals from the document the information is contained in. Hence, it is a fully context-free information importance estimator. We compared our new summarization model with well-known reference systems. The reference systems work, as expected, very well on standard newswire summarization datasets because newswire summarization datasets contain easy-to-exploit importance signals such as information position and information frequency. We artificially removed these signals in further experiments. We first shuffled the sentences to remove the position feature, and we oversampled infrequent information such that all information appears with the same frequencies in the data. In this setting, the performance of all reference systems dropped, whereas the performance of our model remained constant since its importance estimation does not use any of the perturbed importance signals.

Our key conclusion is that available extractive summarization systems share a common weakness: they rely on correlated but not causally linked signals for importance estimation. We showed in our experiments that information position and information frequency are only correlated and not causal by removing the correlation from the dataset without modifying the contained information. Hence, the importance of information in the documents did not change, but the model's beliefs changed, which is not reasonable and cannot happen if the used features have a causal link to information importance. Hence, the only way to build reliable summarization systems is to move beyond simple importance signals to more robust information importance estimators.

Another conclusion that can be drawn is that estimating information importance with information position and information frequency signals do not require any understanding of natural language. We can simply replace meaningful words in a text by arbitrary symbols. Summarization systems that estimate information importance solely based on position and frequency will not change their output. This result shows that today's summarization systems do not have a good language understanding. As we have already argued previously, more challenging datasets, whose summarization requires natural language understanding, are necessary to advance beyond simple summarization systems.

The proposed summarization system can, however, only be considered to be a first prototype for context-free summarization systems. Much more research has to be conducted to build better context-free information importance estimators. One way to build better context-free information importance estimators would be to further leverage available summarization datasets to learn to predict the utility of individual sentences based on more sophisticated sentence representations. A major question for future research is how knowledge can be acquired that can be helpful for information importance estimation. Work in question answering might serve as an inspiration for this research direction.

Not only the development but also the evaluation of context-free and contextual summarization systems can be improved. Due to the lack of available datasets, we evaluated the capabilities of the proposed summarization system on modified newswire summarization datasets. By now, some more heterogeneous datasets such as *h*MDS and auto-*h*MDS are available and can be used to evaluate summarization models.

Context-free importance estimation may also be used to merge single- and multi-document summarization. So far, single- and multi-document summarization are considered to be two different tasks and are only rarely viewed together (Wan, 2010). Usually, multi-document summarization is considered to be an out-of-domain test for summarization models. Since context-free information importance estimation does not use importance signals derived from the source documents, it can be used in both multi- and single-document summarization. In fact, it does not matter in which context information appears since context-free estimation of information importance is independent of the context. Hence, context-free information importance estimators can be applied without modification in both tasks. We already used this fact in the experiments in Chapter 6 in which we used a large single-document summarization dataset to train a context-free information importance estimator and applied it to a multi-document summarization task.

We furthermore resolved a common misconception in a subfield of automatic summarization called sentence regression. The key idea of sentence regression is to estimate utilities for sentences. Most works use the ROUGE$_n$ recall as regressand, which seems reasonable since the final summaries are usually evaluated by computing the ROUGE$_n$ recall of the generated summary. We showed in our experiments that this decision might lead to suboptimal results when greedy selection algorithms are used (which is often the case). We found that it is better to use ROUGE$_n$ precision scores of sentences as regressands instead of the ROUGE$_n$ recall even if the final summary is evaluated with ROUGE$_n$ recall. The reason for this observation is that ROUGE$_n$ recall prefers long sentences in every decision step, which leads to the selection of fewer sentences. No space is wasted because of this bias towards longer sentences if ROUGE$_n$ precision is maximized in every step. For future work, we would like to evaluate if our observation can improve already published greedy sentence regression works.

8.6 How can information importance estimators be evaluated automatically?

Finally, Chapter 7 discussed the evaluation of summarization models, which is similarly important or even more important than learning to estimate information importance. Today's method of choice called ROUGE is based on estimating the similarity between automatically generated summaries and reference summaries created by human annotators. We analyzed that this is problematic for at least two reasons. First, computing the similarity between long texts is a difficult and unsolved problem. Even similarity computation between individual and rather short sentences can not be performed automatically. Hence, the result of the similarity estimation is usually not very reliable. Second, creating reference summaries is very expensive since it requires a lot of human effort and domain expertise, which leads to small and few summarization datasets. However, we have already concluded that large summarization datasets might help to train more advanced summarization models in the future.

To advance evaluation, we developed a new evaluation method that does not rely on reference summaries but cheap pairwise preferences. The key idea is very simple: summaries which contain more important sentences are better. To estimate sentence importance, we collected pairwise preference annotations, which indicate which of two sampled sentences contains more important information. Based on the Bradley-Terry model, we used the pairwise preferences to learned utilities for sentences and used the utilities to estimate the quality of summaries. The proposed method consumes only little time for annotations and does not require the creation of complex summaries. In an experiment, we validated that the prediction performance of the developed method is able to outperform frequently used ROUGE variants. For future work, we would like to improve the developed method by using supervised learning and active learning instead of random sampling to improve sample efficiency. This improvement might reduce the cost and/or further improve the performance of the evaluation model. In addition to the training process, more sophisticated similarity estimations are a possible extension in future work. As discussed in Chapter 3, we would also like to see more evaluation focused on the importance estimation, for example, by evaluating rankings according to importance or by evaluating preference prediction capabilities.

Besides the development of a new evaluation method, we also investigated in detail how evaluation models for summarization should (not) be evaluated. We analyzed previously used evaluation methods and identified limited validity. Due to aggregation effects, very poor evaluation methods can achieve very high correlations with human annotators. Hence, we encouraged to evaluate evaluation models by estimating their pairwise prediction performance, which has been done already by previous work (Owczarzak et al., 2012). We additionally added empirical and theoretical justification of why using pairwise prediction performance is superior to correlation estimation with Pearson's correlation scores.

9 Limitations

In this chapter, we discuss some disadvantages and limitations of the proposed idea of context-free information importance estimation.

9.1 Training Context-free Importance Estimators is Difficult

In this thesis, we presented the idea of context-free information importance estimators which are supposed to estimate the importance of information with common sense knowledge instead of using heuristic importance signals such as information location or information frequency. It is, however, yet unclear if and how models can acquire the required domain knowledge to summarize, for example, the soccer transcript discussed in Chapter 4. We implemented a context-free importance estimator in Chapter 6. The implemented system should, however, rather be considered a prototype. If and how more sophisticated methods can eventually be the method of choice for information importance estimation is subject to future research.

9.2 Domain Knowledge Required

Another limitation of our approach is that we propose to solely use domain knowledge to estimate the importance of information and do not use other potentially correlated importance signals. It has been discussed, however, that also humans sometimes use fast and frugal heuristics for problem-solving (Gigerenzer & Todd, 1999). Furthermore, Kahneman (2012) discusses that humans have two ways to make choices: fast, intuitive thinking, and slow, rational thinking. He explains, for example, that humans tend to believe statements more if they are stated in boldface. Hence, it might be helpful also to consider such simple and maybe not always correct signals for decision making. Instead of only focusing on using domain knowledge to estimate importance, it is perhaps also a good idea to further develop methods that use fast and frugal heuristics and fast decision making. In a situation in which a person has domain knowledge about soccer, for example, but does not have any knowledge about rugby, a person with some common sense reasoning would still be able to summarize a rugby match to some extent.

Nevertheless, we think that it is helpful to not only build complex models that combine multiple strategies for importance estimation but to also explore context-free methods in isolation to better investigate their potentials and limitations.

9.3 Ignoring Author's Intentions

Texts are often written by authors with a particular goal in mind. Journalists, for example, aim at providing new and important information to their readers and structure documents in order to achieve this goal. As discussed above, journalists tend to write more important information at the beginning of an article. Furthermore, paragraphs are used, for example, to signal that a though ends, and a new subtopic starts. The first sentence if each paragraph introduces the new subtopic or can provide a hinge what the reader is going to read in the next few sentences. Context-free methods are not able to catch up on such intentionally created signals since they do not analyze the document structure. In this thesis, we focused on providing the user with the most important information which can be substantially different from the intentions of the author of an article. Methods that also estimate the intentions of the author can complement context-free importance estimators.

10 Final Remarks

Why are humans able to estimate the importance of information to summarize text documents? Why are computers not able to perform this task similarly well? What is the missing piece computers needed to achieve human-level importance estimation skills?

The guiding hypothesis of this thesis is that computers are not capable of human-level information importance estimation because they do not have access to the required common knowledge to truly understand *why* important information is important. They do not know why reporting the winner of the U.S. presidential election is more important than reporting details about the certification of the results. We claim that computers will never achieve human-level information importance estimation skills as long as they rely on simple heuristics derived from the information's context. Sometimes information importance correlates well with information position or information frequency. However, this correlation is not causally linked to the importance of information. Information is not important because it appears first in a document, and information is also not important because it frequently appears in a document. Information is important because it has an impact on people's life.

Therefore, let us further explore context-free information importance estimation!

List of Definitions

List of Figures

List of Tables

Bibliography

Agrawal, A., Batra, D., & Parikh, D. (2016). Analyzing the Behavior of Visual Question Answering Models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 1955–1960).

Aharon, R. B., Szpektor, I., & Dagan, I. (2010). Generating Entailment Rules from FrameNet. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 241–246).

Alonso, L., Castellón, I., Casas, B., & Padró, L. (2004). Knowledge-intensive automatic e-mail summarization in CARpAntA. In *Proceedings of the ACL 2004 on Interactive poster and Demonstration Sessions* (pp. 16–19).

Arumae, K., & Liu, F. [Fei]. (2018). Reinforced Extractive Summarization with Question-Focused Rewards. In *Proceedings of ACL 2018, Student Research Workshop* (pp. 1–7).

Attokurov, U., & Bayazit, U. (2014). Multi-document summarization using distortion-rate ratio. *Proceedings of the ACL 2014 Student Research Workshop*, 64–70.

Baker, C. F. [Collin F.], Fillmore, C. J., & Lowe, J. B. (1998). The Berkeley FrameNet Project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics* (pp. 86–90).

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence* (pp. 2670–2676).

Bar-Hillel, Y. (1969). Wesen und Bedeutung der Informationstheorie. In H. von Ditfurth (Ed.), *Informationen über Information* (pp. 13–42). Hamburg: Hoffmann und Campe.

Bar-Hillel, Y., & Carnap, R. (1953). Semantic Information. *The British Journal for the Philosophy of Science*, *4*(14), 147–157.

Bär, D., Zesch, T., & Gurevych, I. (2013). DKPro Similarity: An Open Source Framework for Text Similarity. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 121–126.

Benikova, D., Mieskes, M., Meyer, C. M., & Gurevych, I. (2016). Bridging the gap between extractive and abstractive summaries: Creation and evaluation of coherent extracts from heterogeneous sources. In *Proceedings of the 22nd International Conference on Computational Linguistics* (pp. 1039–1050).

Bergius, R. (1971). *Psychologie des Lernens*. De Gruyter.

Bloem, P., Mota, F., de Rooij, S., Antunes, L., & Adriaans, P. (2014). A Safe Approximation for Kolmogorov Complexity. *Algorithmic Learning Theory*, *8776*, 336–350.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008). Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data* (pp. 1247–1250).

Bradley, R. A., & Terry, M. E. (1952). Rank Analysis of Incomplete Block Designs. *Biometrika*, *39*(3), 324–345.

Brandow, R., Mitze, K., & Rau, L. F. (1995). Automatic condensation of electronic publications by sentence selection. *Information Processing and Management*, *31*(5), 675–685.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. (1984). *Classification and Regression Trees*. Taylor & Francis Ltd.

Brin, S., & Page, L. (2012). Reprint of: The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, *56*(18), 3825–3833.

Cao, Z., Wei, F., Dong, L., Li, S., & Zhou, M. (2015). Ranking with recursive neural networks and its application to multi-document summarization. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence* (pp. 2153–2159).

Cao, Z., Wei, F., Li, S., Li, W., Zhou, M., & Wang, H. (2015). Learning Summary Prior Representation for Extractive Summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (pp. 829–833).

Carbonell, J., & Goldstein, J. (1998). The use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *Proceedings of the 21st Annual ACM SIGIR International Conference on Research and Development in Information Retrieval* (pp. 335–336).

Celikyilmaz, A., & Hakkani-Tur, D. (2011). Discovery of Topically Coherent Sentences for Extractive Summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics* (pp. 491–499).

Chen, W., Liu, T.-y., Lan, Y., Ma, Z.-m., & Li, H. (2009). Ranking Measures and Loss Functions in Learning to Rank. In *Advances in Neural Information Processing Systems 22* (pp. 315–232).

Cheng, J., & Lapata, M. (2016). Neural Summarization by Extracting Sentences and Words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (pp. 484–494). Berlin, Germany.

Cheung, J. C. K., & Penn, G. (2013). Towards Robust Abstractive Multi-Document Summarization : A Caseframe Analysis of Centrality and Domain. In *Proceedings of the 51st Annual Meeting ofthe Association for Computational Linguistics* (pp. 1233–1242).

Chopra, S., Auli, M., & Rush, A. M. (2016). Abstractive Sentence Summarization with Attentive Recurrent Neural Networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 93–98).

Chou, P. A., Lookabaugh, T., & Gray, R. M. (1989). Optimal Pruning with Applications to Tree-Structured Source Coding and Modeling. *IEEE Transactions on Information Theory*, *35*(2), 299–315.

Christensen, J., Mausam, Soderland, S., Etzioni, O., Mausam, Soderland, S., & Etzioni, O. (2013). Towards Coherent Multi-Document Summarization. *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (Section 3), 1163–1173.

Christensen, J., Soderland, S., & Bansal, G. (2014). Hierarchical Summarization: Scaling Up Multi-Document Summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* (pp. 902–912).

Cleveland, A. D., & Cleveland, D. B. (2013). *Introduction to Indexing and Abstracting* (4th Editio). Libraries Unlimited.

Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S., Chang, W., & Goharian, N. (2018). A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 615–621).

Cohan, A., & Goharian, N. (2016). Revisiting Summarization Evaluation for Scientific Articles. In *Proceedings of the 10th International Conference on Language Resources and Evaluation* (pp. 806–813).

Cohn, T., & Lapata, M. (2008). Sentence Compression Beyond Word Deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics* (pp. 137–144).

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing. In *Proceedings of the 25th International Conference on Machine Learning* (pp. 160–167).

Colmenares, C. A., Litvak, M., & Sheva, B. (2015). HEADS : Headline Generation as Sequence Prediction Using an Abstract Feature-Rich Space of Engineering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (pp. 133–142).

Conroy, J. M., & O'leary, D. P. (2001). Text summarization via hidden Markov models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 406–407).

Cook, S. (2000). The P vs NP problem. In *The Millennium Prize Problem*, Clay Mathematical Institute.

Dang, H. T. (2005). Overview of DUC 2005.

Dang, H. T. (2006). Overview of Document Understanding Conference 2006, National Institute of Standards and Technology (NIST).

Dang, H. T. (2007). Overview of Document Understanding Conference 2007, National Institute of Standards and Technology (NIST).

Dang, H. T., & Owczarzak, K. (2008). Overview of the TAC 2008 Update Summarization Task. In *Proceedings of the 1st Text Analysis Conference* (pp. 1–16). Gaithersburg, Maryland, USA: National Institute of Standards and Technology (NIST).

Dang, H. T., & Owczarzak, K. (2009). Overview of the TAC 2009 Summarization Track. In *Proceedings of the Second Text Analysis Conference*, Gaithersburg, Maryland, USA.

Dernoncourt, F., Ghassemi, M., & Chang, W. (2018). A Repository of Corpora for Summarization. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference* (pp. 3221–3227).

Donaway, R. L., Drummey, K. W., & Mather, L. A. (2000). A comparison of rankings produced by summarization evaluation measures. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic Summarization* (Vol. 4, pp. 69–78).

Durrett, G., Berkeley, U. C., Berg-kirkpatrick, T., Klein, D., & Berkeley, U. C. (2016). Learning-Based Single-Document Summarization with Compression and Anaphoricity Constraints. In *Proceedings of the 54th Annual Meeting ofthe Association for Computational Linguistics* (pp. 1998–2008).

Eckart de Castilho, R., & Gurevych, I. (2014). A broad-coverage collection of portable NLP components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT* (pp. 1–11).

Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the Association for Computing Machinery*, *16*(2), 264–285.

Endres, D. M., & Schindelin, J. E. (2003). A New Metric for Probability Distributions. *IEEE Transactions on Information Theory*, *49*(7), 1858–1860.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, *22*, 457–479.

Faccio, D., Clerici, M., & Tambuchi, D. (2006). Revisiting the 1888 Hertz experiment. *American Journal of Physics*, *74*(11), 992–994.

Falke, T., & Gurevych, I. (2017). Bringing Structure into Summaries: Crowdsourcing a Benchmark Corpus of Concept Maps. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 2951–2961).

Falke, T., Meyer, C. M., & Gurevych, I. (2017). Concept-Map-Based Multi-Document Summarization using Concept Coreference Resolution and Global Importance Optimization. In *Proceedings of the 8th International Joint Conference on Natural Language Processing* (pp. 801–811).

Filatova, E., & Hatzivassiloglou, V. (2004). A formal model for information selection in multi-sentence extraction. In *Proceedings of the 20th International Conference on Computational Linguistics* (pp. 397–403).

Filippova, K., & Altun, Y. (2013). Overcoming the Lack of Parallel Data in Sentence Compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 1481–1491).

Fillmore, C. J. (1976). Frame Semantics and the Nature of Language. *Annals of the New York Academy of Sciences*, *280*(1), 20–32.

Fillmore, C. J., & Baker, C. F. [Collin F]. (2001). Frame semantics for text understanding. In *Proceedings of WordNet and Other Lexical Resources Workshop* (pp. 3–4).

Fisher, R. (1925). Theory of statistical estimation. *Proceedings of the Cambridge Philosophical Society*, *8*, 700–725.

Flexner, A. (1939). The usefulness of useless knowledge. *Harper's Magazine*, (179), 544–552.

Floridi, L. (2010). *Information - A Very Short Introduction*. Oxford University Press.

Floridi, L. (2011). *The Philosophy of Information*. Oxford University Press.

Fürnkranz, J., & Hüllermeier, E. (Eds.). (2010). *Preference Learning*. Springer.

Gambhir, M., & Gupta, V. (2017). Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review*, *47*(1), 1–66.

Ganesan, K., Zhai, C., & Han, J. (2010). Opinosis : A Graph-Based Approach to Abstractive Summarization of Highly Redundant Opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics* (pp. 340–348).

Gao, D., Li, W., & Zhang, R. (2013). Sequential Summarization: A New Application for Timely Updated Twitter Trending Topics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (pp. 567–571).

Gernert, D. (2006). Pragmatic Information: Historical Exposition and General Overview. *Mind & Matter*, *4*(2), 141–167.

Giannakopoulos, G. (2013). Multi-document multilingual summarization and evaluation tracks in ACL 2013 MultiLing Workshop. In *Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization* (p. 20).

Giannakopoulos, G., Conroy, J. M., Kubina, J., Rankel, P. A., Lloret, E., Litvak, M., & Favre, B. (2017). MultiLing 2017 Overview. In *Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres* (pp. 1–6).

Giannakopoulos, G., & Karkaletsis, V. (2013). Summary Evaluation: Together We Stand NPowER-ed. In *Proceedings of the 14th International Conference on Computational Linguistics and Intelligent Text Processing* (pp. 436–450).

Giannakopoulos, G., Kubina, J., Conroy, J. M., Steinberger, J., Favre, B., Kabadjov, M., ... Poesio, M. (2015). MultiLing 2015: Multilingual Summarization of Single and Multi-Documents, On-line Fora, and Call-center Conversations. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue* (pp. 270–274).

Gibbs, J. W. (1906). *The scientific papers of J. Willard Gibbs in Two Volumes. Volume 1: Thermodynamics*. Longmans, Green, and Co.

Gigerenzer, G., & Todd, P. M. (1999). *Simple Heuristics That Make Us Smart*. Oxford University Press.

Gildea, D., & Jurafsky, D. (2002). Automatic Labeling of Semantic Roles. *Computational Linguistics*, *28*(3), 245–288.

Gillick, D., Favre, B., & Hakkani-Tür, D. (2008). The ICSI Summarization System at TAC 2008. In *Proceedings of the Text Analysis Conference*.

Gillick, D., Favre, B., Hakkani-Tür, D., Bohnet, B., Liu, Y., & Xie, S. (2009). The ICSI/UTD Summarization System at TAC 2009. In *Proceedings of the 2nd Text Analysis Conference*.

Goldstein, J., Kantrowitz, M., Mittal, V., & Carbonell, J. (1999). Summarizing Text Document: senetence Selection and Evaluation Metrics. *Proceedings of the 22nd Annual International ACM SIGIR Cnference on Research and Development in Information Retrieval*, 121–128.

Graham, Y. (2015). Re-evaluating Automatic Summarization with BLEU and 192 Shades of ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 128–137).

Grusky, M., Naaman, M., & Artzi, Y. (2018). Newsroom: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies. In *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 708–719).

Gupta, S., Nenkova, A., & Jurafsky, D. (2007). Measuring importance and query relevance in topic-focused multi-document summarization. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions* (p. 193).

Habernal, I., Sukhareva, M., Raiber, F., Shtok, A., Kurland, O., Ronen, H., ... Gurevych, I. (2016). New Collection Announcement. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 701–704).

Hartley, R. (1928). Transmission of Information. *Bell System Technical Journal*, *7*(3), 535–563.

Hartmann, S., Kuznetsov, I., Martin, T., & Gurevych, I. (2017). Out-of-domain FrameNet Semantic Role Labeling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 471–482).

Heinzerling, B., Judea, A., & Strube, M. (2015). HITS at TAC KBP 2015: Entity discovery and linking, and event nugget detection. In *Proceedings of the Text Analysis Conference*.

Herbrich, R., Minka, T., & Graepel, T. (2006). TrueSkill: A Bayesian Skill Rating System. In *Advances in Neural Information Processing Systems 20* (pp. 569–576).

Hermann, K., Kocisky, T., & Grefenstette, E. (2015). Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 29* (pp. 1693–1701).

Hilgard, E. (1948). *Theories of Learning*. Appleton-Century-Crofts.

Hochbaum, D. S. (1997). Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. In *Approximation Algorithms for NP-hard Problems* (pp. 94–143). PWS Publishing Co.

Hong, K., Conroy, J. M., Favre, B., Kulesza, A., Lin, H., & Nenkova, A. (2014). A Repository of State of the Art and Competitive Baseline Summaries for Generic News Summarization. In *Proceedings of the 9th International Conference on Language Resources and Evaluation* (pp. 1608–1616).

Hong, K., & Nenkova, A. (2014). Improving the Estimation of Word Importance for News Multi-Document Summarization. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 712–721).

Hovy, E. (2005). Text Summarization. *The Oxford Handbook of Computational Linguistics*, 583–598.

Hovy, E., & Lin, C.-Y. (1999). Automated text summarization in SUMMARIST. In I. Mani & M. T. Maybury (Eds.), *Advances in Automatic Text Summarization* (Chap. 8, pp. 81–94). MIT Press.

Hsu, W.-T., Lin, C.-K., Lee, M.-Y., Min, K., Tang, J., & Sun, M. (2018). A Unified Model for Extractive and Abstractive Summarization using Inconsistency Loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (pp. 1–10).

Hu, B., Chen, Q., & Zhu, F. (2015). LCSTS: A Large Scale Chinese Short Text Summarization Dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 1967–1972).

Hu, M., & Liu, B. (2004). Mining and Summarizing Customer Reviews. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Huang, T.-K., Lin, C.-j., & Weng, R. C. (2006). Ranking individuals by group comparisons. In *Proceedings of the 23rd International Conference on Machine learning* (Vol. 9, pp. 425–432).

Huang, X., Wan, X., & Xiao, J. (2011). Comparative news summarization using linear programming. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (pp. 648–653).

Itti, L., & Baldi, P. (2005). Bayesian Surprise Attracts Human Attention Laurent. In *Advances in Neural Information Processing Systems 18* (pp. 1295–1306).

Jadhav, A. (2018). Extractive Summarization with SWAP-NET : Sentences and Words from Alternating Pointer Networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (pp. 1–10).

Janssen, F., & Fürnkranz, J. (2010). On the quest for optimal rule learning heuristics. *Machine Learning*, *78*(3), 343–379.

Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, *20*(4), 422–446.

Jia, R., & Liang, P. (2017). Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 2021–2031).

Jing, H., Barzilay, R., McKeown, K. R., & Elhadad, M. (1998). Summarization evaluation methods: Experiments and analysis. *AAAI Symposium on Intelligent Summarization*, 51–59.

Jing, H., & McKeown, K. R. (1999). The decomposition of human-written summary sentences. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 129–136.

Kågebäck, M., Mogren, O., Tahmasebi, N., & Dubhashi, D. (2014). Extractive Summarization using Continuous Vector Space Models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality* (pp. 31–39).

Kahneman, D. (2012). *Thinking, Fast and Slow*. Penguin.

Kant, I. (1800). *Logik – ein Handbuch zu Vorlesungen*.

Kedzie, C., McKeown, K., & Diaz, F. (2015). Predicting salient updates for disaster summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (pp. 1608–1617).

Kendall, M. G. (1938). A New Measure of Rank Correlation. *Biometrika*, *30*(1/2), 81.

Kendall, M. G., & Gibbons, J. (1990). *Rank Correlation Methods* (5th Editio). Oxford University Press.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the Association for Computing Machinery*, *46*(5), 604–632.

Knight, K., & Marcu, D. (2002). Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, *139*(1), 91–107.

Kohlschütter, C., Fankhauser, P., & Nejdl, W. (2010). Boilerplate detection using shallow text features. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining* (pp. 441–450).

Kullback, S., & Leibler, R. A. (1951). On Information and Sufficiency. *Annals of Mathematical Statistics*, *22*(1), 79–86.

Kupiec, J., Pedersen, J., & Chen, F. (1995). A Trainable Document Summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 68–73).

Li, C., Qian, X., & Liu, Y. (2013). Using Supervised Bigram-based ILP for Extractive Summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (pp. 1004–1013).

Li, M., & Vitányi, P. (2008). *An Introduction to Kolmogorov Complexity and Its Applications*. Springer New York.

Li, S., Ouyang, Y., & Sun, B. (2006). Peking University at DUC 2006. In *Proceedings of the Document Understanding Conference 2006*.

Li, S., Ouyang, Y., Wang, W., & Sun, B. (2007). Multi-document summarization using support vector regression. *Proceedings of the Document Understanding Conference 2007*.

Li, W., Wu, M., Lu, Q., Xu, W., & Yuan, C. (2006). Extractive Summarization using Inter-and Intra-Event Relevance. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL* (pp. 369–376).

Likert, R. (1932). A technique for the measurement of attitudes. *Archives of Psychology*, *22 140*, 55.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out* (pp. 25–26).

Lin, C.-Y., & Hovy, E. (2000). The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th Conference on Computational linguistics* (pp. 495–501).

Lin, C.-Y., & Hovy, E. (2002a). From Single to Multi-document Summarization: A Prototype System and its Evaluation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 457–464).

Lin, C.-Y., & Hovy, E. (2002b). Manual and automatic evaluation of summaries. In *Proceedings of the Workshop on Automatic Summarization* (Vol. 4, *July*, pp. 45–51).

Lin, C.-Y., & Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology* (pp. 71–78).

Lin, H., & Bilmes, J. (2011). A Class of Submodular Functions for Document Summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics* (pp. 510–520).

Ling, X., & Weld, D. S. (2012). ine-Grained Entity Recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence* (pp. 94–100).

Liu, F. [Feifan], & Liu, Y. (2008). Correlation between ROUGE and Human Evaluation of Extractive Meeting Summaries. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (pp. 201–204).

Liu, M., Li, W., Wu, M., & Lu, Q. (2007). Extractive summarization based on event term clustering. In *Proceedings of the ACL 2007 Demo and Poster Sessions* (pp. 185–188).

Liu, P. J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., & Shazeer, N. (2018). Generating Wikipedia By Summarizing Long Sequences. In *Proceedings of the 6th International Conference on Learning Representations* (pp. 1–14).

Lloret, E., Plaza, L., & Aker, A. (2013). Analyzing the capabilities of crowdsourcing services for text summarization. *Language Resources & Evaluation*, *47*(2), 337–369.

Louis, A. (2014). A Bayesian Method to Incorporate Background Knowledge during Automatic Text Summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* (pp. 333–338).

Louis, A., & Nenkova, A. (2013). Automatically Assessing Machine Summary Content Without a Gold Standard. *Computational Linguistics*, *39*(2), 267–300.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, *2*(2), 159–165.

Ly, A., Marsman, M., Verhagen, J., Grasman, R. P., & Wagenmakers, E. J. (2017). A Tutorial on Fisher information. *Journal of Mathematical Psychology*, *80*, 40–55.

Mackie, S., Mccreadie, R., Macdonald, C., & Ounis, I. (2014). On Choosing an Effective Automatic Evaluation Metric for Microblog Summarisation. In *Proceedings of the 5th Information Interaction in Context Symposium* (pp. 115–124).

Mani, I. (2001). *Automatic Summarization*. John Benjamins Publishing Co.

Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT Press.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (pp. 55–60).

Marshak, J. (1954). Towards an Economic Theory of Organization and Information. In *Economic Information, Decision, and Prediction: Selected Essays: Volume II* (pp. 29–62). Springer Netherlands.

Mausam, Schmitz, M., Bart, R., Soderland, S., & Etzioni, O. (2012). Open Language Learning for Information Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 523–534).

Meade, F. (1997). Using Robust NLP and Machine Learning. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics* (pp. 62–66).

Mihalcea, R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions* (4, pp. 20–23).

Mihalcea, R. (2005). Language Independent Extractive Summarization. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions* (pp. 49–52).

Mihalcea, R., & Tarau, P. (2004). TextRank: Bringing order into texts. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing* (Vol. 85, pp. 404–411).

Mihaylov, T., & Frank, A. (2016). Discourse Relation Sense Classification Using Cross-argument Semantic Similarity Based on Word Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (pp. 100–107).

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26* (pp. 3111–3119).

Mullis, I. V., Kennedy, A. M., Martin, M. O., & Sainsbury, M. (2006). *PIRLS 2006 - Assessment framework and specifications*. TIMSS & PIRLS International Study Center.

Nallapati, R., Zhai, F., & Zhou, B. (2017). SummaRuNNer: A Recurrent Neural Network based Sequence Model for Extractive Summarization of Documents. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence* (pp. 3075–3081).

Nallapati, R., Zhou, B., dos Santos, C. N., Gulcehre, C., & Xiang, B. (2016). Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning* (pp. 280–290).

Napoles, C., Gormley, M., & Durme, B. V. (2012). Annotated Gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction & Web-scale Knowledge Extraction* (pp. 95–100).

Nemhauser, G. L., Wolsey, L. A., & Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions-I. *Mathematical Programming*, *14*(1), 265–294.

Nenkova, A., & McKeown, K. (2011). Automatic Summarization. *Foundations and Trends in Information Retrieval*, *5*(3), 103–233.

Nenkova, A., & Passonneau, R. (2004). Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 145–152).

Nenkova, A., Passonneau, R., & McKeown, K. (2007). The Pyramid Method. *ACM Transactions on Speech and Language Processing*, *4*(2), 1–23.

Nenkova, A., & Vanderwende, L. (2005). The Impact of Frequency on Summarization. In *Technical Report*, Microsoft Research.

Ng, J.-P., Bysani, P., Lin, Z., Kan, M.-Y., & Tan, C.-L. (2012). Exploiting Category-Specific Information for Multi-Document Summarization. In *Proceedings of the 24th International Conference on Computational Linguistics* (pp. 2093–2108).

Ng, J.-P., Chen, Y., Kan, M.-Y., & Li, Z. (2014). Exploiting timelines to enhance multi-document summarization. In *Proceedings of the 52nd Annual Meeting ofthe Association for Computational Linguistics* (Vol. 1, pp. 923–933).

Nomoto, T., & Matsumoto, Y. (2001a). A new approach to unsupervised text summarization. In *Proceedings of the 24th Annual International ACM SIGIR conference on Research and Development in Information Retrieval* (pp. 26–34).

Nomoto, T., & Matsumoto, Y. (2001b). Supervised ranking in open-domain text summarization. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (p. 465).

Ouyang, Y., Li, W., & Lu, Q. (2009). An Integrated Multi-document Summarization Approach Based on Word Hierarchical Representation. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, (August), 113–116.

Ouyang, Y., Li, W., Li, S., & Lu, Q. (2011). Applying regression models to query-focused multi-document summarization. *Information Processing and Management*, *47*(2), 227–237.

Over, P. (2001). Introduction to DUC-2001: An Intrinsic Evaluation of Generic News Text Summarization Systems, National Institute of Standards and Technology (NIST).

Over, P., Dang, H., & Harman, D. (2007). DUC in context. *Information Processing and Management*, *43*(6), 1506–1520.

Over, P., & Liggett, W. (2002). Introduction to DUC-2002: An Intrinsic Evaluation of Generic News Text Summarization Systems, National Institute of Standards and Technology (NIST).

Over, P., & Yen, J. (2003). Introduction to DUC-2003: An Intrinsic Evaluation of Generic News Text Summarization Systems, National Institute of Standards and Technology (NIST).

Over, P., & Yen, J. (2004). Introduction to DUC-2004: An Intrinsic Evaluation of Generic News Text Summarization Systems, National Institute of Standards and Technology (NIST).

Owczarzak, K., Conroy, J. M., Dang, H. T., & Nenkova, A. (2012). An assessment of the accuracy of automatic evaluation in summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization* (pp. 1–9).

Owczarzak, K., & Dang, H. T. (2010). Overview of the TAC 2010 Summarization Track. In *Proceedings of the 3rd Text Analysis Conference*, Gaithersburg, Maryland, USA.

Owczarzak, K., & Dang, H. T. (2011). Overview of the TAC 2011 Summarization Track. In *Proceedings of the 4th Text Analysis Conference*, Gaithersburg, Maryland, USA.

P. V. S., A., & Meyer, C. M. (2017). Joint Optimization of User-desired Content in Multi-document Summaries by Learning from User Feedback. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (pp. 1353–1363).

P. V. S., A., Peyrard, M., & Meyer, C. M. (2018). Live Blog Corpus for Summarization. In *Proceedings of the 11th International Conference on Language Resources and Evaluation* (pp. 3197–3203).

Paice, C. D. (1990). Constructing Literature Abstracts by Computer: Techniques and Prospects. *Information Processing and Management*, *26*(1), 171–186.

Papadimitriou, C. H. (1981). On the Complexity of Integer Programming. *Journal of the Association for Computing Machinery*, *28*(4), 765–768.

Papineni, K., Roukos, S., Ward, T., & Zhu, W. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 311–318).

Parker, R., Graff, D., Kong, J., Chen, K., & Maeda, K. (2011). *English Gigaword Fifth Edition LDC2011T07*. Linguistic Data Consortium.

Parveen, D., & Strube, M. (2015). Integrating importance, non-redundancy and coherence in graph-based extractive summarization. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence* (pp. 1298–1304).

Passonneau, R. J., Chen, E., Guo, W., & Perin, D. (2013). Automated Pyramid Scoring of Summaries using Distributional Semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (pp. 143–147).

Paulus, R., Xiong, C., & Socher, R. (2018). A Deep Reinforced Model for Abstractive Summarization. In *Proceedings of the 6th International Conference on Learning Representations* (pp. 1–13).

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (pp. 1532–1543).

Peyrard, M., & Eckle-Kohler, J. (2017). Supervised Learning of Automatic Pyramid for Optimization-Based Multi-Document Summarization. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 1084–1094.

Pilehvar, M. T., Jurgens, D., & Navigli, R. (2013). Align, Disambiguate and Walk: A Unified Approach for Measuring Semantic Similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (pp. 1341–1351).

Pitler, E., Louis, A., & Nenkova, A. (2004). Automatic Evaluation of Linguistic Quality in Multi-Document Summarization. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 544–554.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., & Webber, B. (2008). The Penn Discourse TreeBank 2.0. *Proceedings of the 6th International Conference on Language Resources and Evaluation*, 2961–2968.

Radev, D. R., Allison, T., Blair-Goldensohn, S., Blitzer, J., Celebi, A., Dimitrov, S., . . . Liu, D., et al. (2004). MEAD-A Platform for Multidocument Multilingual Text Summarization. In *Proceedings of the 4th International Conference on Language Resources and Evaluation* (pp. 1–4).

Radev, D. R., Hovy, E., & McKeown, K. (2002). Introduction to the special issue on summarization. *Computational Linguistics*, *28*(4), 399–408.

Radev, D. R., Jing, H., & Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Proceedings of the 2000*

NAACL-ANLP Workshop on Automatic Summarization (pp. 21–30). Association for Computational Linguistics.

Radev, D. R., Tam, D., & Erkan, G. (2003). Single-document and multi-document summary evaluation using Relative Utility. *Proceedings of the 2003 ACM CIKM International Conference on Information and Knowledge Management*, 1–28.

Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2383–2392.

Ramond, P. (2010). Group Theory. *Physics*, *455*(October).

Rashkin, H., Singh, S., & Choi, Y. (2016). Connotation Frames: A Data-Driven Investigation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (pp. 311–321).

Ren, P., Wei, F., & Chen, Z. (2016). A Redundancy-Aware Sentence Regression Framework for Extractive Summarization. In *Proceedings of the 26th International Conference on Computational Linguistics* (pp. 33–43).

Rush, A. M., Chopra, S., & Weston, J. (2015). A Neural Attention Model for Abstractive Sentence Summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 379–389).

Saggion, H., & Lapalme, G. (2002). Generating indicative-informative summaries with SumUM. *Computational Linguistics*, *28*(4), 497–526.

Sandhaus, E. (2008). *The New York Times Annotated Corpus LDC2008T19*. Linguistic Data Consortium.

Schluter, N., & Søgaard, A. (2015). Unsupervised extractive summarization via coverage maximization with syntactic and semantic concepts. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (2009, pp. 840–844).

See, A., Liu, P. J., & Manning, C. D. (2017). Get To The Point : Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (pp. 1073–1083).

Shang, G., Ding, W., Zhang, Z., Tixier, A. J.-P., Meladianos, P., Vazirgiannis, M., & Lorré, J.-P. (2018). Unsupervised Abstractive Meeting Summarization with Multi-Sentence Compression and Budgeted Submodular Maximization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (pp. 1–11).

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(July 1928), 379–423.

Shannon, C. E., & Weaver, W. (1964). *The mathematical theory of communication*. Urbana, Illinois: The University of Illinois Press.

Sjöbergh, J. (2007). Older versions of the ROUGEeval summarization evaluation system were easier to fool. *Information Processing and Management*, *43*(6), 1500–1505.

Socher, R., Huang, E. H., Pennington, J., Ng, A. Y., & Manning, C. D. (2011). Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection. In *Advances in Neural Information Processing Systems 24* (pp. 1–9).

Sparck Jones, K. (1999). Automatic summarising: factors and directions. In *Advances in Automatic Text Summarisation* (pp. 1–12). MIT Press.

Spiliopoulou, E., Hovy, E., & Mitamura, T. (2017). Event Detection Using Frame-Semantic Parser. In *Proceedings of the Events and Stories in the News Workshop* (pp. 15–20).

Tauchmann, C., Arnold, T., Hanselowski, A., Meyer, C. M., & Mieskes, M. (2018). Beyond Generic Summarization: A Multi-faceted Hierarchical Summarization Corpus of Large Heterogeneous Data. *Proceedings of the 11th International Conference on Language Resources and Evaluation*, 3184–3191.

Tixier, A. J., Malliaros, F. D., & Vazirgiannis, M. (2016). A graph degeneracy-based approach to keyword extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 1860–1870).

Tixier, A., Skianis, K., & Vazirgiannis, M. (2016). GoWvis: A Web Application for Graph-of-Words-based Text Visualization and Summarization. In *Proceedings of the 54th Annual Meeting ofthe Association for Computational Linguistics: System Demonstrations* (pp. 151–156).

Torres-Moreno, J.-M. (2014). *Automatic Text Summarization*.

Toutanova, K., & Brockett, C. (2016). A Dataset and Evaluation Metrics for Abstractive Compression of Sentences and Short Paragraphs. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 340–350).

Verberne, S., Krahmer, E., Hendrickx, I., Wubben, S., & van Den Bosch, A. (2018). Creating a reference data set for the summarization of discussion forum threads. *Language Resources & Evaluation*, *52*(2), 461–483.

von Weizsäcker, C. F. (1985). *Aufbau der Physik*. Munich: Deutscher Taschenbuch Verlag.

Wan, X. (2010). Towards a Unified Approach to Simultaneous Single-Document and. *Proceeding of the 23rd International Conference on Computational Linguistics*, (August), 1137–1145.

Wan, X., Yang, J., & Xiao, J. (2007). Towards an Iterative Reinforcement Approach for Simultaneous Document Summarization and Keyword Extraction. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (pp. 552–559).

Wang, D., Zhu, S., & Li, T. (2009). Multi-document summarization using sentence-based topic models. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP* (pp. 297–300).

Wang, Y., Wang, L., Li, Y., He, D., Chen, W., & Liu, T.-Y. (2013). A Theoretical Analysis of NDCG Ranking Measures. In *JMLR Workshop and Conference Proceedings* (pp. 1–30).

Wasson, M. (1998). Using leading text for news summaries. In *Proceedings of the 36th Annual Meeting on Association for Computational Linguistics* (pp. 1364–1368).

Weaver, W. (1949). The Mathematics of Communication. *Scientific American*, *181*(1), 11–15.

Weinberger, E. D. (2002). A theory of pragmatic information and its application to the quasi-species model of biological evolution. *BioSystems*, *66*(3), 105–119.

Wilkins, D. E., Lee, T. J., & Berry, P. (2003). Interactive execution monitoring of agent teams. *Journal of Artificial Intelligence Research*, *18*, 217–261.

Yang, Q., Passonneau, R. J., & de Melo, G. (2016). PEAK: Pyramid Evaluation via Automated Knowledge Extraction. In *Proceedings of the 30th Conference on Artificial Intelligence* (pp. 2673–2679).

Yao, J.-g., Wan, X., & Xiao, J. (2017). Recent advances in document summarization. *Knowledge and Information Systems*, *53*(2), 297–336.

Ye, S., Chua, T.-S., & Lu, J. (2009). Summarizing definition from Wikipedia. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP* (pp. 199–207).

Yih, W. T., Goodman, J., Vanderwende, L., & Suzuki, H. (2007). Multi-document summarization by maximizing informative content-words. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence* (pp. 1776–1782).

Yin, W., & Pei, Y. (2015). Optimizing sentence modeling and selection for document summarization. In *The Proceedings of the 24th International Joint Conference on Artificial Intelligence* (pp. 1383–1389).

Zermelo, E. (1929). Die Berechnung der Turnier-Ergebnisse als ein Maximumproblem der Wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*, *29*, 436–460.

Zhong, Y. (2017). A theory of semantic information. *China Communications*, *14*(1), 1–17.

Zhou, L., Hovy, E., & Rey, M. (2005). Digesting Virtual "Geek" Culture : The Summarization of Technical Internet Relay Chats. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics* (pp. 298–305).

Zintgraf, L. M., Cohen, T. S., Adel, T., & Welling, M. (2017). Visualizing Deep Neural Network Decisions: Prediction Difference Analysis. In *arXiv preprint* (pp. 1–12).

Zopf, M. (2015). SeqCluSum: Combining Sequential Clustering and Contextual Importance Measuring to Summarize Developing Events over Time. In *Proceedings of the 24th Text Retrieval Conference*.

Zopf, M. (2018a). auto-hMDS: Automatic Construction of a Large Heterogeneous Multi-Document Summarization Corpus. In *Proceedings of the 11th International Conference on Language Resources and Evaluation* (pp. 3228–3233).

Zopf, M. (2018b). Estimating Summary Quality with Pairwise Preferences. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1687–1696).

Zopf, M., Botschen, T., Falke, T., Heinzerling, B., Marasovi, A., Fürnkranz, J., & Frank, A. (2018). What's important in a text ? An extensive evaluation of linguistic annotations for summarization. In *Fifth International Conference on Social Networks Analysis, Management and Security* (pp. 272–277).

Zopf, M., Loza Mencía, E., & Fürnkranz, J. (2016a). Beyond Centrality and Structural Features: Learning Information Importance for Text Summarization. In *Proceedings of the 20th Conference on Computational Natural Language Learning* (pp. 84–94).

Zopf, M., Loza Mencía, E., & Fürnkranz, J. (2016b). Sequential Clustering and Contextual Importance Measures for Incremental Update Summarization. In *Proceedings of the 26th International Conference on Computational Linguistics* (pp. 1071–1082).

Zopf, M., Loza Mencía, E., & Fürnkranz, J. (2018). Which Scores to Predict in Sentence Regression for Text Summarization? In *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1782–1791).

Zopf, M., Peyrard, M., & Eckle-Kohler, J. (2016). The Next Step for Multi-Document Summarization : A Heterogeneous Multi-Genre Corpus Built with a Novel Construction Approach. In *Proceedings of the 26th International Conference on Computational Linguistics* (pp. 1535–1545).

A Notes on Research Data Management

We provide in this chapter additional information about the research data management with respect to the research conducted to generated this thesis. Further information about research data management provided by the Deutsche Forschungsgemeinschaft (DFG, English: German Research Foundation) can be found in the corresponding 'Guidelines for Research Data Management'.[1] In the context of this thesis, research data includes in particular generated scientific papers, data, and software.

Important parts of this thesis have been published at scientific conferences and are available online. This includes:

1. Zopf (2018b): Published by the Association for Computational Linguistics, licensed under a Creative Commons Attribution 4.0 License,[2] and available online in the ACL Anthology at http://www.aclweb.org/anthology/N18-1152

2. Zopf, Loza Mencía, and Fürnkranz (2018): Published by the Association for Computational Linguistics, licensed under a Creative Commons Attribution 4.0 License, and available online in the ACL Anthology at http://www.aclweb.org/anthology/N18-1161

3. Zopf, Botschen, et al. (2018): Published by the Institute of Electrical and Electronics Engineers and available online in the IEEE Xplore Digital Library at https://ieeexplore.ieee.org/document/8554853

4. Zopf (2018a): Published by the European Language Resources Association, licensed under a Creative Commons Attribution-NonCommercial 4.0 International License,[3] and available online at http://www.lrec-conf.org/proceedings/lrec2018/pdf/1018.pdf

5. Zopf, Peyrard, and Eckle-Kohler (2016): Published by the International Committee on Computational Linguistics, licensed under a Creative Commons Attribution 4.0 License, and available online and available online in the ACL Anthology at http://www.aclweb.org/anthology/C16-1145

6. Zopf, Loza Mencía, and Fürnkranz (2016b): Published by the International Committee on Computational Linguistics, licensed under a Creative Commons Attribution 4.0 License, and available online and available online in the ACL Anthology at http://www.aclweb.org/anthology/C16-1102

7. Zopf, Loza Mencía, and Fürnkranz (2016a): Published by the Association for Computational Linguistics, licensed under a Creative Commons Attribution 4.0 License, and available online and available online in the ACL Anthology at http://www.aclweb.org/anthology/K16-1009

[1] http://www.dfg.de/download/pdf/foerderung/antragstellung/forschungsdaten/richtlinien_forschungsdaten.pdf
[2] https://creativecommons.org/licenses/by/4.0/
[3] https://creativecommons.org/licenses/by-nc/4.0/

8. Zopf (2015): Published by the National Institute of Standards and Technology and available online at https://trec.nist.gov/pubs/trec24/papers/AIPHES-TS.pdf

Datasets used or created in this thesis are mainly corpora containing text document. This includes:

1. Document Understanding Conference: Data from the DUC shared tasks is available upon request according to the AQUAINT Information-Retrieval Text Research Collections User Agreement[4] and the TIPSTER Information-Retrieval Text Research Collections User Agreement.[5]

2. Text Analysis Conference: Data from the TAC 2008 and TAC 2009 shared tasks[6] is available upon request according the the AQUAINT-2 User Agreement.[7] The AQUAINT-2 collection is a subset of the Linguistic Data Consortium English Gigaword Third Edition which is archived under catalog number LDC2007T07.[8]

3. TREC Temporal Summarization: Data from the TREC Temporal Summarization shared task is a subset of the TREC StreamCorpus generated for the KBA track and is available online.[9] The corresponding encryption key is available upon request according to TREC KBA Information-Retrieval Text Research Collections User Agreement.[10]

4. hMDS: Data of the created hMDS corpus (see Section 5.3) is available online in Github at https://github.com/AIPHES/hMDS. This includes guidelines used to create the corpus as well as detailed description of the corpus. The reference documents have been extracted from Wikipedia. The respective versions are archived by the Wikimedia Foundation Inc. and available online under a Creative Commons Attribution-ShareAlike 3.0 Unported License.[11] The source document have been archived in the Internet Archive.[12] A link list pointing to the corresponding web pages is available upon request. The corpus has furthermore been archived internally. Due to copyright reasons, it is not possible to make the data publicly available.

5. auto-hMDS: Data of the created auto-hMDS corpus (see Section 5.5) is available online in Github at https://github.com/AIPHES/auto-hMDS. This includes a list of all topics included in the corpus as well as all raw and preprocessed reference summaries. The summaries are archived and licensed by the Wikimedia Foundation Inc. under a Creative Commons Attribution-ShareAlike 3.0 Unported License. A link list pointing to the corresponding source document web pages is available upon request. The corpus has furthermore been archived internally. Due to copyright reasons, it is not possible to make the data publicly available.

[4] https://www-nlpir.nist.gov/projects/duc/forms/org_appl_aquaint.html

[5] https://www-nlpir.nist.gov/projects/duc/forms/org_appl_tips.html

[6] https://tac.nist.gov

[7] https://tac.nist.gov/data/data_desc.html#AQUAINT-2

[8] https://catalog.ldc.upenn.edu/LDC2007T07

[9] http://s3.amazonaws.com/aws-publicdatasets/trec/kba/index.html

[10] https://trec.nist.gov/data/kba/TREC-KBA-Organizational-User-Agreement.html

[11] https://creativecommons.org/licenses/by-sa/3.0/legalcode

[12] https://archive.org

Furthermore, software that has been used to run experiments is archived in a version control system of the Knowledge Engineering Group, TU Darmstadt[13]. This includes in particular code to create corpora for Chapter 5, code to run summarization experiments in Chapter 6, and code to run evaluation experiments in Chapter 7.

[13] http://www.ke.tu-darmstadt.de