Personalization of Hearing Aids using Daily Routine Recognition and Sensor Fusion

Vom Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt

zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation von

Thomas Kübert, M. Sc.

Erstgutachter: Prof. Dr.-Ing. Henning Puder
Zweitgutachter: Prof. Dr. techn. Heinz Koeppl

Tag der Einreichung: 24.11.2021
Tag der mündlichen Prüfung: 29.03.2022

D17
Darmstadt 2022

Thomas Kübert: Personalization of Hearing Aids
using Daily Routine Recognition and Sensor Fusion
Darmstadt, Technische Universität Darmstadt,
Jahr der Veröffentlichung der Dissertation auf TUprints: 2022
URN: urn:nbn:de:tuda-tuprints-214265
Tag der mündlichen Prüfung: 29.03.2022

Acknowledgments

I am grateful to everyone for their contributions to the success of this dissertation.

First of all, I would like to express my deepest appreciation to my supervisor, Prof. Dr.-Ing. Henning Puder, during the period of doctoral candidacy and master thesis. Prof. Puder gave me the opportunity to pursue my PhD at the Signal Processing Group of Sivantos GmbH in Erlangen. I am extremely grateful to your guidance, interest in my work and support at any time. You always provided me a valuable feedback and your fast proactive approach solved the upcoming problems ahead of time. This efficient working style had a strong influence on me.

Furthermore, I would like to extend my deepest gratitude to my co-supervisor, Prof. Dr. techn. Heinz Koeppl. Thank you for also providing me a valuable feedback and bringing in new ideas with your open mindset. I would like to thank you to invite me to the Bioinspired Communication Systems group. A special thanks goes to the members, Bastian Alt and Lukas Koehs.

I also would like to thank the members of my committee: Prof. Dr. rer. nat. Andy Schürr, Prof. Dr.-Ing. Christoph Hoog Antink, and Prof. Dr. rer. nat. Florian Steinke. Many thanks to the colleagues at the Graduate School of Computational Engineering at TU Darmstadt, where I was an associated member during the time of doctoral candidacy. In particular, the annual retreat, soft-skill seminars, and the research cycle broaden my horizon with inspiring talks and new approaches. Thereby, I enjoyed the discussions and company of the members: Afief Dias Pambudi, Gerta Kushe, Dario Klingenberg, Iryna Kulchytska-Ruchka, Mona Fuhrländer, Mehdi Elasmi, Alexander Birx, Anna Melina Merkel, and Jan Becker. A special thanks goes to the current and former staff members at the Graduate School, namely Dr. Markus Lazanowski, Dr.-Ing. Melanie Gattermayer, Christian Schmitt, Steffi Vass, and Carina Schuster. I also would like to thank the members of university, namely Ann-Kathrin Seifert, Tim Schaeck, Patricia Binder, Freweyni Kidane Teklehaymanot, Patrick Wenzel, and Sara Al-Sayed.

I would like to extend my sincere thanks to Sivantos GmbH for funding and further support, which made this work possible. In particular, I cannot begin to express my thanks to Jana Thiemt and Megha Subramanian, who extended the mobile application for my study and developed the special hearing aid firmware to support the Bluetooth data transmission. I very much appreciate my colleagues from the signal processing and further departments. It is a pleasure working with Bernd Meister, Dr.-Ing. Dirk Mauler, Dr.-Ing. Tobias Rosenkranz, Dr.-Ing. Tobias Wurzbacher, Dr.-Ing. Alberto Escalante Banuelos, Dr.-Ing. Stefan Petrausch, Dr. Homayoun Kamkar Parsi, Eghart Fischer, Robert Kasanmascheff, Dr.-Ing. Marko Lugger, Ju-

liane Borsum, Jens Hain, Oliver Dressler, Dr.-Ing. Stefan Wehr, Nina Sebald, Benjamin Graf, Johannes Scheitacker, Dr.-Ing. Dianna Yee, Dr.-Ing. Mehdi Zohourian, Dr.-Ing. Michael Buerger, Pascal Zobel, Hendrik Barfuss, Ion Suberviola, Wolfgang Soergel, Thomas Pilgrim, Dr.-Ing. Ronny Hannemann, Dr.-Ing. Dirk Junius, Dr. Maja Serman, PhD Karolina Smeds, Florian Wolters, Dr. Nadja Schinkel-Bielefeld, Dr. Heike Heuermann, Sebastian Best, Frank Loch, Uwe Flaig, Tom Maennel, David Wolfart, Chris Hesse, Alastair Manders, Sascha Bilert, Nicola Acs, Anja Knoll, Bene Heuer, Kristina Hunger, Ilse Ramirez, Katharina Mohnlein, Sebastian Brauer, Max Schmitt, Sandro Kecanociv, Julia Warmuth, Kaja Kallisch, and all other not explicitly mentioned. You always had an open ear and helpful hand. A special thanks goes to Maria Schmitt for the support in administrative, traveling, and billing issues.

I would like to express my special thanks to all study participants for their efforts during the long recordings. Without your time and efforts, this research work would not be possible. Furthermore, I would like to thank all current and former student members of Sivantos GmbH. Particularly helpful to me during this time were Ann-Kristin Seifer, Maximiliane Hawesch, Franz Wichert, Nico Kaiser, Johannes Zeitler, Maximilian Mayer, Christoph Daube, Fauzy Dawwara, Jonas Baetcke, Metehan Yurt, Patricia Kunz, Alina Mosebach, Julia Zimmer, Maximilian Kerner, Justin Cauthen, Samet Rasih Koeseli, Cathrina Veigel, Fabian Hettler, Jakob Rippel, Florian Hilgemann, Carmen Tluczykont, and Jana Welling.

Finally, I am deeply grateful to my family and friends, and express my appreciation to my parents, Maria and Manfred, and my brother, Torsten. Their continuous support enabled me to develop my skills for my academic career and I got the person I am. In addition, I would like to thank my girlfriend, Catharina, which always supported, encouraged me to withstand my doubts and overcome the challenges. This support and all the others helped me to achieve my goals and successfully complete this dissertation.

Thank you!

Erlangen, November 2021.

Zusammenfassung

Laut der Weltgesundheitsorganisation sind weltweit über 400 Millionen Menschen von der Schwerhörigkeit betroffen. Moderne Hörgeräte (HA) können diese Einschränkungen des Hörvermögens reduzieren. Dies erfordert aber eine situationsabhängige Steuerung, die die eingehenden Geräusche personenunabhängig in vordefinierte akustische Kategorien, wie z.B. Sprache in Lärm, einteilt. Für jede Kategorie existiert eine entsprechende Einstellung, z.B. die Frequenzverstärkung. Jedoch zeigt die komplexe Audiosignalverarbeitung den höchsten Nutzen, wenn die Algorithmen optimal an die jeweiligen akustischen Situationen und persönlichen Vorlieben angepasst sind. Für eine natürliche und subtile HA-Kontrolle ist eine stabile und zuverlässige Situationsidentifikation notwendig. Zur Erhöhung der Benutzerzufriedenheit und Steigerung der zeitlichen Prädiktionsstabilität schlagen wir die Personalisierung des Klassifizierungssystems für jeden Benutzer vor, indem die wiederkehrenden Situationen und Umgebungen der täglichen Routine berücksichtigt werden. Dazu verknüpfen wir die täglichen Routinesituationen und -umgebungen mit bevorzugten HA-Einstellungen. Daher schlagen wir als Erste eine Kombination aus Beschleunigungs- (ACC) und Mikrofondaten zur Erkennung der täglichen Routine in Hörgeräten vor.

Während akustische Klassifizierungssysteme typischerweise auf ausgewählten realen und kontrollierten Situationen trainiert werden, führen wir unsere Analysen ausschließlich in realistischen, unbeschränkten Situationen und Umgebungen von HA-Trägern durch, die ihrem persönlichen Tagesablauf folgen. Daher erstellen wir zwei realistische große Datensätze, die die Grundlage für unsere umfassenden Untersuchungen bilden. Der erste Datensatz $\mathbb{D}_{\mathrm{T}}$ besteht aus einem Probanden über 9 Tage mit groben Tagebuchannotationen für unsere Machbarkeitsexperimente. Der zweite Datensatz $\mathbb{D}_7$ umfasst sieben Probanden über 104 Tage mit absichtsbasierten Benutzerannotationen für die Modellgeneralisierungsuntersuchungen. Die Aufnahmen basieren auf einem HA-Prototypen, der die Übertragung der ACC- und Audiodaten auf ein Mobiltelefon ermöglicht.

Um die Realisierbarkeit des Ansatzes aufzuzeigen und zu analysieren, welche Situationen innerhalb des Merkmalraums unterscheidbar sind, führen wir Clustering- und Visualisierungsansätze am Datensatz $\mathbb{D}_{\mathrm{T}}$ durch. Damit zeigen wir, dass die ACC- und Audio-Merkmale verschiedene Alltagssituationen und -umgebungen unterscheiden und gruppieren können. Diese werden visualisiert und gruppiert in Datenembeddings und Merkmalsgraphen über die Zeit. Mit dem Wissen um die unterscheidbaren Routinesituationen und die groben Tagebuchannotationen beschriften wir den Datensatz $\mathbb{D}_{\mathrm{T}}$ durch einen erweiterten semi-überwachten Algorithmus. Danach werden die Routineaktivitäten erkannt und der Einfluss von drei Eingangsvarianten,

nämlich ACC, Audio und die Kombination aus beiden Sensoren, analysiert. In diesen Experimenten zeigen wir den starken Beitrag der Audiomerkmale.

Nach der Darstellung des Proof-of-Concept wird der zweite Datensatz $\mathbb{D}_7$ verwendet, um die Modellgeneralisierungsfähigkeiten über Probanden hinweg zu analysieren. Wir trainieren mehrere Offline-, Online- und Sequenzklassifikatoren. Hierfür bauen wir einen effizienten Merkmalsraum auf, der die wiederkehrenden täglichen Situationen und Umgebungen gut beschreibt. Zur Erkennung des Tagesablaufs wenden wir verschiedene Offline-Klassifizierungsmethoden an und testen diese. Das mehrschichtige Perzeptron und der Random Forest (RF), die personenabhängig trainiert wurden, zeigen die besten Ergebnisse im F-Maß. Wir bestätigen mit unserer Analyse für die High-Level-Aktivitäten, dass das personenabhängige Modell das unabhängige Modell im direkten Vergleich zueinander übertrifft. Das Ziel unserer Online-Experimente ist die Verbesserung der Offline-Klassifizierungsergebnisse und die Simulation eines realen Systems. Für diesen Zweck personalisieren wir ein Modell, das personenunabhängig vortrainiert wurde, durch tägliche Online-Updates mit den vorhergesagten oder wahren Benutzerannotationen. Dabei werden mehrere inkrementelle Lernansätze und ein Online-RF getestet. Wir zeigen, dass der RF das F-Maß im Vergleich zu den Offline-Baselines selbst verbessern kann. In unseren Sequenzsimulationen modellieren wir die zeitlichen Beziehungen der sequentiellen Daten, um die Routineerkennung zu optimieren. Die Ergebnisse zeigen eine starke Verbesserung der täglichen Routineerkennung mit den vorgeschlagenen Sequenzklassifikationstechniken. Die Sequenzansätze erhöhen die zeitliche Stabilität der Vorhersagen.

In dieser Dissertation zeigen wir das Potential einer personalisierten täglichen Routineklassifizierung für eine optimale HA-Konfiguration auf. Unser effizientes Verarbeitungsschema ermöglicht die Erkennung der Routineklassen, die mit einer bevorzugten HA-Einstellung verbunden sind, basierend auf Audio- und Beschleunigungsmerkmalen. Unsere Arbeit unterstützt Hörgeräteträger mit Hilfe eines verbesserten Klassifizierungssystems, um die Benutzerzufriedenheit zu erhöhen.

Abstract

According to the World Health Organization, disabling hearing loss affects over 400 million people worldwide. Modern hearing aids (HA) can reduce this burden but require a situation-dependent control, which classifies the incoming sounds in a person-independent manner into predefined acoustic categories, such as speech in noise. For each category a corresponding setting, e.g., frequency gains, exists. However, the complex audio signal processing shows the highest benefits if the algorithms are optimally adapted to the respective acoustic situations and personal preferences. A stable and reliable situation identification is necessary for a natural and subtle HA control. To enhance the user satisfaction and increase the temporal prediction stability, we propose to personalize the classification system for each user by considering the recurring situations and environments of the daily routine. For this purpose, we link the daily routine situations and environments to preferred HA settings. Therefore, we are first to propose a combination of acceleration (ACC) and microphone data to recognize the daily routine in hearing aids.

While acoustic classification systems are typically trained on selected real and controlled situations, we solely perform our analysis on realistic unconstrained situations and environments of HA wearers following their personal daily routine. Therefore, we create two realistic large data sets that form the basis for our comprehensive investigations. The first set $\mathbb{D}_{\mathrm{T}}$ contains one subject over 9 days with coarse time diary annotations for our feasibility experiments. The second data set $\mathbb{D}_7$ includes seven subjects over 104 days with intention-based user annotations for the model generalization investigations. For the recordings, we build on a HA prototype allowing to stream the ACC and audio data to a mobile phone.

To show the feasibility of the approach and analyze which situations are distinguishable within the feature space, we perform clustering and visualization approaches on the data set $\mathbb{D}_{\mathrm{T}}$. Thereby, we demonstrate that the ACC and audio features discriminate and group various daily routine situations and environments. These are visualized and clustered in data embeddings and feature plots over time. Using the knowledge of the discriminative routine situations and the coarse time diary annotations, we label the $\mathbb{D}_{\mathrm{T}}$ set by an extended semi-supervised algorithm. After that, the routine activities are recognized and the effect of three input variants, namely ACC, audio, and both, are analyzed. Within these experiments, we show the strong contribution of the audio features.

After showing the proof-of-concept, the second set $\mathbb{D}_7$ is used to analyze the model generalization abilities across subjects. We train several classifiers in an offline, online, and sequence manner. To achieve this, we build an efficient feature repre-

sentation, which describes the recurring daily situations and environments well. To recognize the daily routine, we apply and test various classification methods in an offline manner. The multi-layer perceptron and random forest (RF) trained in a person-dependent way show the best F-measure performance. We confirm for high-level activities that the person-dependent model outperforms the independent one. In our online experiments, the goal is to improve the offline classification results and simulate a real system. Therefore, we personalize a model that was pretrained in a person-independent manner by daily online updates with the predicted or true user labels. Thereby, multiple incremental learners and an online RF are tested. We demonstrate that the RF can self-improve the F-measure compared to the offline baselines. In our sequence simulations, we model the temporal relationships of the sequential data to improve the routine detection. The results show a strong improvement in the daily routine recognition with the proposed sequence classification techniques. The sequence learners enhance the temporal stability of the predictions.

In this dissertation, we show the potential of a personalized daily routine classification for an optimal HA configuration. Our efficient processing scheme allows to detect the routine classes linked to a preferred HA setting based on audio and acceleration features. To this end, our work contributes to support hearing aid wearers with an enhanced classification system to improve the user satisfaction.

Contents

1 Introduction **1**
- 1.1 Supporting Hearing Impaired People with Hearing Aids . . . 1
- 1.2 Adapting Hearing Aids to the Acoustic Scene by Person-Independent Classification . . . 1
- 1.3 Improving Scene Classification by Personalization and Daily Routine Recognition . . . 2
- 1.4 Specifying Objectives of the Thesis . . . 3
- 1.5 Summarizing the State of the Art and Contributions . . . 5
  - 1.5.1 Creating Two Daily Routine Data Sets and Labels . . . 5
  - 1.5.2 Grouping and Visualizing the Daily Routine Data by Unsupervised Learning . . . 7
  - 1.5.3 Labeling the Daily Routine Data by Semi-Supervised Learning 8
  - 1.5.4 Features for Daily Routine Recognition . . . 9
  - 1.5.5 Classifying the Daily Routine Data by Supervised Learning . . 9
  - 1.5.6 Adapting the Classification Models by Online Learning . . . 10
  - 1.5.7 Modeling the Temporal Daily Routine Transitions by Sequence Learning . . . 11
- 1.6 Summarizing the Publications . . . 12
- 1.7 Structuring the Thesis . . . 13

2 Daily Routine Data Sets and Labels **15**
- 2.1 Common Hearing Situations and Labels for Classification System . . 15
- 2.2 Intentional Daily Routine Labels . . . 17
- 2.3 Data Sets and Recording Procedure . . . 18
  - 2.3.1 Data Set of Thomas $\mathbb{D}_{\mathrm{T}}$ . . . 19
  - 2.3.2 Data Set of Seven Subjects $\mathbb{D}_{7}$ . . . 20
  - 2.3.3 Data Set of Huynh $\mathbb{D}_{\mathrm{H}}$ . . . 23
- 2.4 Description of Hearing Aid Features . . . 25
  - 2.4.1 Acceleration Signal Model . . . 26
  - 2.4.2 Audio Features . . . 29
  - 2.4.3 Feature Notation . . . 30
- 2.5 Data Summary . . . 30

3 Grouping and Visualizing the Daily Routine Data by Unsupervised Learning **33**
- 3.1 Feature Extraction for Clustering . . . 33
- 3.2 Clustering the Daily Routine Data . . . 34
  - 3.2.1 Clustering Techniques . . . 35

3.2.2 Clustering Evaluation Metrics . . . 37
3.3 Visualizing the Daily Routine Data Temporally . . . 39
3.4 Visualizing the Daily Routine Data Spatially by Manifold Learning . 40
3.4.1 Dimensionality Reduction and Manifold Learning Techniques . 40
3.4.2 Evaluating the best Dimensionality Reduction and Manifold Learning Technique . . . 43
3.5 Confirming the Clustering and Manifold Assumptions . . . 45
3.6 Grouping and Visualization Summary . . . 46

4 Labeling the Daily Routine Data by Semi-Supervised Learning 49
4.1 Solving the Labeling Problem . . . 49
4.1.1 Labeling Techniques . . . 49
4.1.2 Visual Interactive Labeling Technique . . . 50
4.1.3 Visual Interactive Labeling Results . . . 52
4.2 Comparing the Daily Routine Recognition Rates by Different Input Features . . . 53
4.2.1 Classification Techniques . . . 55
4.2.2 Classification Evaluation and Experimental Setup . . . 56
4.2.3 Classification Results . . . 56
4.3 Labeling Summary . . . 58

5 Classifying the Daily Routine Data by Supervised Learning 59
5.1 Feature Extraction and Selection Techniques for Classification . . . 59
5.1.1 Low-Level Sensor Fusion . . . 60
5.1.2 High-Level Feature Extraction and Selection . . . 62
5.2 Classification Techniques . . . 63
5.3 Classification Evaluation and Experimental Setup . . . 68
5.4 Classification Results . . . 69
5.5 Classification Summary . . . 71

6 Improving the Daily Routine Classification by Modeling the Temporal Behavior 73
6.1 Adapting the Classification Models by Online Learning . . . 73
6.1.1 Online Learning Techniques . . . 73
6.1.2 Online Learning Evaluation and Experimental Setup . . . 79
6.1.3 Online Learning Results . . . 80
6.2 Modeling the Temporal Daily Routine Transitions by Sequence Learning 84
6.2.1 Sequence Learning Techniques . . . 84
6.2.2 Sequence Learning Evaluation and Experimental Setup . . . 87
6.2.3 Sequence Learning Results . . . 89
6.3 Improved Classification Summary . . . 96

7 Conclusions and Outlook 97
7.1 Conclusions . . . 97
7.2 Outlook . . . 100

Annex **105**
A.1 Data Recording Study Protocol and Manual for the $\mathbb{D}_7$ set 105
A.2 Proposed Audiological Optimization of Personalized Routine Classes . 105
A.3 Integration of Personalized Daily Routine and Person-Independent Acoustic Classification 107
A.3.1 Fusion of Personalized and General Decision Making 107
A.3.2 Fusion of Hearing Aid Settings from Person-Independent and Personalized Classification 108

Abbreviations, Acronyms, and Symbols **109**

List of References **113**

Chapter 1

Introduction

1.1 Supporting Hearing Impaired People with Hearing Aids

Hearing impairment negatively affects a gradually increasing number of people worldwide. According to the World Health Organization, over 400 million humans suffer from disabling hearing loss and these people represent over 5% of the world's population [1]. These high numbers are expected to raise until the year 2050 over 900 million people. For adults, a hearing loss of greater than 40 decibels (dB) is called a disabling hearing loss compared to the better hearing ear. Various reasons for this condition exist, such as genetic causes, ear infections, exposure to heavy noise, or aging. In particular, elderly people over 65 years are highly affected with an approximated share of one third. The negative impact of hearing loss results in difficulties during conversations, in particular, in loud environments, localizing sounds in traffic situations, and typical activities like watching television [2]. Thus, the social interactions are more complicated due to a difficult speech understanding, which is frustrating for the impaired people. To help them, hearing aids (HA) can at least partially compensate the hearing loss. The device can apply a frequency-selective amplification to the incoming sounds, directional processing, e.g., to focus the front of the impaired person and reduce other directions, and noise reduction measures to weaken, for example, impulse or wind noise [3]. These measures improve the speech intelligibility and reduce the environmental noise. For the highest benefits, the measures need to be optimally controlled to the current acoustic situation.

1.2 Adapting Hearing Aids to the Acoustic Scene by Person-Independent Classification

To adapt state-of-the-art hearing aids to the current acoustic scene, a person-independent classification is performed on sound features and the system is typically trained on selected real-world and laboratory situations [4]. These recognized classes are predefined categories, such as music, noise, or speech in noise [5]. Afterwards, based on these situation categories a set of predefined algorithmic and audiological parameters is applied to the hearing aids [3]. In this way, the HA adapt to its surroundings and this process is continuously repeated. Thus, in a speech in noise situation the HA can activate the directional processing to emphasize speak-

ers in front of the wearer. In contrast to a music scene, where the amplified higher frequencies and dynamics are preferable for an optimal listening comfort [2].

Unfortunately, the acoustic scene is highly non-stationary, changes in short-time intervals and can be ambiguous [6]. Due to this behavior, the acoustic classification can lead to frequent unwanted setting changes. These modifications of HA parameters, e.g., frequency-dependent gains, can be uncomfortable for the wearer. Therefore, a stable and reliable situation identification is necessary for a natural and subtle HA control. Then, the complex audio signal processing shows the highest benefits if the algorithms are optimally adapted to the respective acoustic situations.

1.3 Improving Scene Classification by Personalization and Daily Routine Recognition

To improve the scene classification and the hearing aid parameter adaptation, we propose to personalize the HA configuration and recognize the daily routine by a person-dependent classification. This personalized system is trained on the personal real-world routine situations. In that way, it ensures the temporal stability and takes into account the user behavior and recurring acoustic situations. The daily routine consists of periodic recurring situations and environments plus slowly changes over time. It is a high-level activity, which is a composition of many low-level activities. Thus, we link the common, repetitive situations of the daily routine to a preferred hearing aid setting.

To illustrate an exemplary daily routine, a working day and the corresponding time schedule of a week with a strong repetitive structure are displayed in Fig. 1.1. The working day starts with the morning routine activities, such as daily hygiene and breakfast, plus going to work. The central element is the office work interrupted by the lunch break. Afterwards, commuting back home, which is followed by dinner and other free time activities. That is why, the daily structure often has a certain sequence of events, which can be exploited to identify the scenes based on the temporal dependencies. Elderly people particularly have a strong daily routine [7]. Thus, HA wearers repeatedly face similar situations and environments, which is illustrated in the time schedule in Fig. 1.1 (b). Therefore, we can leverage this knowledge for an improved classification and configuration.

The ideal HA device setting is specified by the user's intention in a certain situation, which translates to different hearing needs. Assuming that the user does office work and the colleague besides him has a conversation with a visitor. Here, due to spatial proximity, the HA would decide that the user wants to listen to this conversation based on the short-term acoustic cues. However, the wearer's intention is to focus on his work. Hence, the audio information can be ambiguous, and we need to consider the user's behavior, e.g., the body movements particularly analyzing the head motions [6], over a longer period to deduce this kind of situations.

Another example would be that the user plays football and someone close to him shouts some commands. Thus, the classification system could decide based on the short-term acoustic cues, that the wearer is in a conversation and activates a directional processing to emphasize this voice. However, the user wants to monitor his total surroundings. Hence, the short-term acoustic cues can be ambiguous, and additional data inputs need to be considered over a longer period, e.g., the motion

behavior, to gain more reliable scene information. Therefore, we use an acceleration sensor within a HA for a better scene analysis. Supporting this new concept of a personalized HA scene adaptation, we focus on the daily routine detection part to intelligently control the HA parameters by taking advantage of the recurring situations and environments. To reach the proposed goal, we specify objectives for this dissertation in the next section.

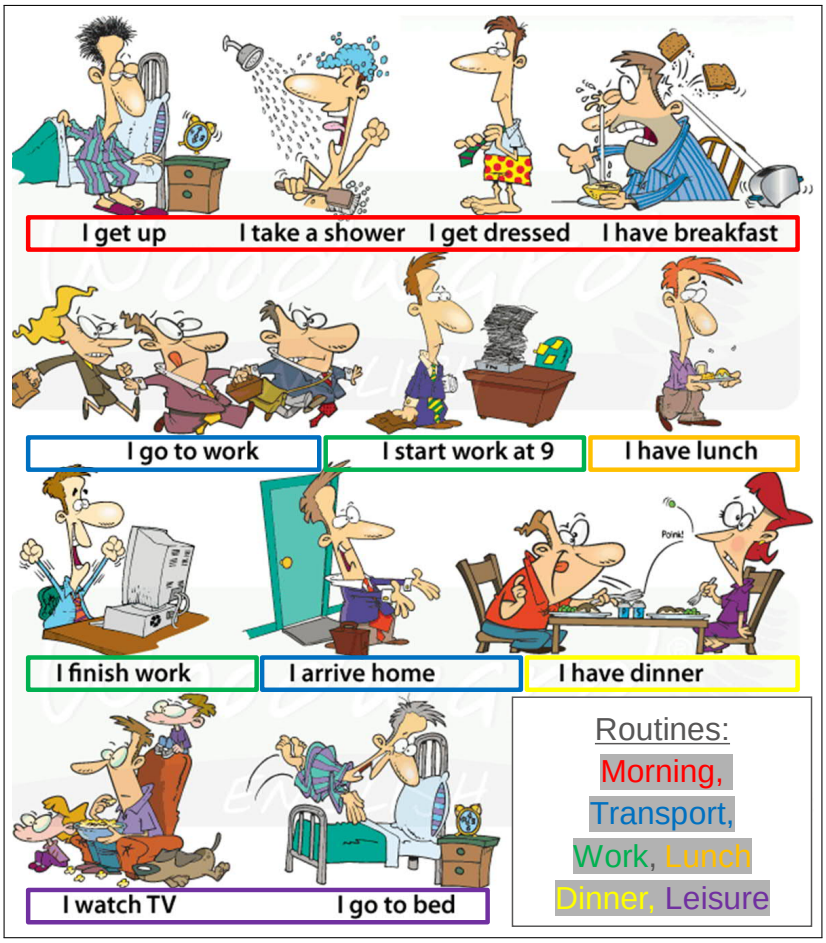
1.4 Specifying Objectives of the Thesis

The primary objective of this dissertation is to improve the classification system of a hearing aid by exploiting the recurring daily routine situations and environments. This enhances the temporal prediction stability and takes into the account the personal situations. Based on this goal, several steps are specified and need to be successfully completed. Firstly, relevant features to distinguish different routine situations should be found. Based on these features, large recordings of the daily routine should be done. After that, these data sets allow to investigate which situations cluster in the feature space and judge the relevance of these features. Then, the detection of these routine situations can be evaluated. Finally, for each situation an optimal audiology setup could be found.

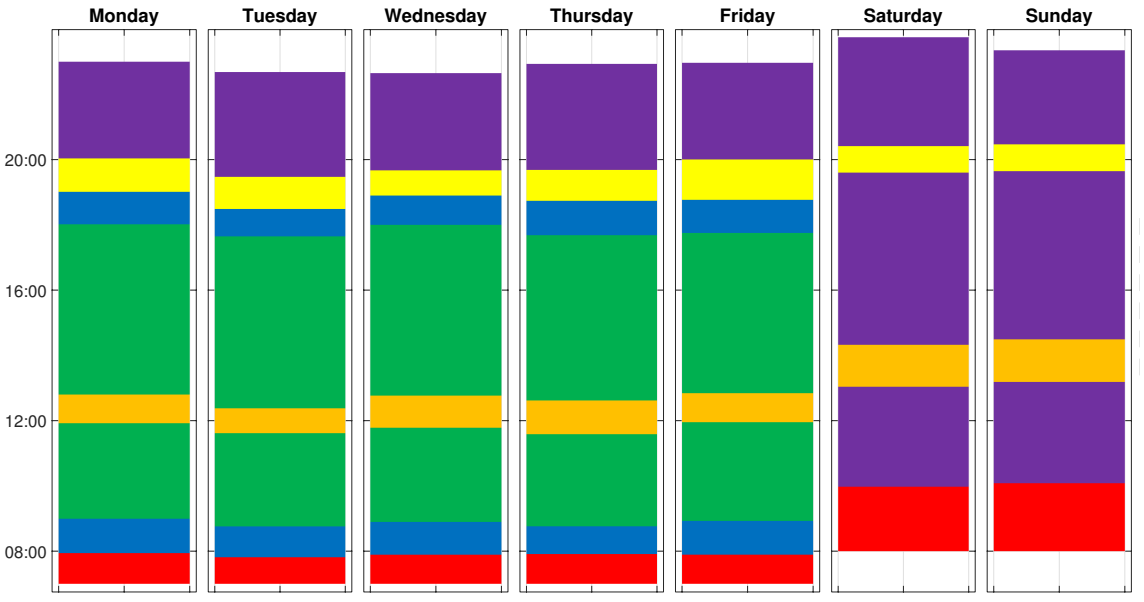
For the identification of recurring characteristic situations, a set of **relevant features** must be found, which describes the scene properties well. Obvious candidates are the acoustic characteristics of the HA, such as the loudness level, own voice detection, features derived from the audio signal in time or frequency domain, acoustic class decisions, such as car, music, signal in noise, or quiet. In addition, in each prototype HA a built-in acceleration (ACC) sensor tracks the head and body motion of the user as a combination of linear translational and rotational movements. These features are transmitted and stored on a connected cell phone, which can upload the sensor outputs to a cloud database for further processing. Furthermore, the mobile phone itself has a rich set of candidate sensors, for example, GPS, light, accelerometer, gyroscope, barometer, and magnetometer.

Based on a first large **data set** of real-life situations, the relevance of the selected features could be assessed if and how well they group the different situations. This could show the feasibility of the approach, meaning the proof of concept. After that, a large data set of multiple subjects in the range of two weeks per person could be recorded to test the recognition performance of characteristic situations across the different subjects. Here, a specific focus can be put on the detection of reoccurring individual characteristic and general situations. In particular, a link to the daily routine can be set up, where, e.g., within the working week a certain pattern of activities happens.

The characteristic scenes can then be grouped based on the inherent similarity between the recurring situations by, e.g., an unsupervised **clustering** approach. This method can be combined by a dimensionality reduction technique to find a low dimensional representation of these scenes. Within this feature space, it would be possible to identify unknown situations for one user, due to similarities between the subjects and scenes, which have already been learned by the data of another subject. Furthermore, developing an easy and intuitive **visualization** of the data can help to capture the daily routine patterns. This representation can use the output of the dimensionality reduction and clustering stage.



(a) The daily routine comic (adapted from [8]).



(b) Exemplary daily routine time schedule.

Figure 1.1: Exemplary daily routine comic shows a representative sequence of events for a workday with a set of recurring situations and environments. The behavior is repeated during the week and at the weekend more leisure activities take place.

The **detection** of these characteristic scenes builds another main target of this dissertation. Therefore, **classification** approaches can be evaluated based on a suitable feature representation and the impact of these features can be assessed. On the basis of these experiments, the model generalization abilities across subjects and days can be assessed to check the personal and daily data variability. Furthermore, different training procedures, namely a personalized or person-independent model training, should be tested. Additionally, fixed offline classification models can be compared against adaptive online methods or sequence models. The online classifiers can simulate a real system that is adapted on new incoming data. The sequence approaches could model the temporal relationship of the routine data.

After the detection of a characteristic situation, the personal preferences and intentions, corresponding to an *optimal HA configuration* for each person, should be evaluated and found. For this optional audiological task, a solution could be proposed based on existing concepts in the literature.

1.5 Summarizing the State of the Art and Contributions

In the following, we review the state of the art per covered topic and summarize the corresponding contributions in this dissertation, where the explained material is partially published in [9, 10, 11]. The written publications are summarized in the next section 1.6, the dependencies of covered topics are explained in section 1.7 and are illustrated in Fig. 1.2. Several topics have been covered and a number of contributions have been achieved during the time of the doctoral candidacy. The main topic areas fall into the daily routine recognition (DRR) and analysis. The topics and achievements group into the creation of two large real-world data sets, the clustering and visualization analysis of routine situations, the feature impact evaluation of acceleration and audio data, the offline, online, and sequence classification investigations.

1.5.1 Creating Two Daily Routine Data Sets and Labels

In the field of human activity recognition (HAR), several data sets exist, and overviews of these sets can be found in [12, 13, 14]. These studies recorded a single person to multiple participants performing low-level activities, such as walking, running, sitting, or cycling, or activities of daily living (ADL, e.g., grooming, food preparation, or cleaning). A few popular examples of HAR data sets are listed in Table 1.1 and we give further details to one of them. The Opportunity data set concentrates on ADL with lots of primitive activities in a sensor-rich and constrained environment [15]. Thereby, researchers often used inertial measurement units (IMU) consisting of a gyroscope, a magneto- and accelerometer, where, in particular, accelerometers were selected due to the low power consumption like in our case. The sensors were placed on the whole body as a wearable unit, e.g., at the legs, arms, or waist, or a worn smartphone with an IMU and more sensors in the trouser pocket was utilized. The interesting head position for the hearing aid research was less often used. One example is a head-mounted IMU sensors to detect the viewing direction for an improved position estimation during walking [16].

Table 1.1: Overview of popular public HAR and audio data sets.

HAR Sets	Activities
Opportunity [15]	ADL
van Kasteren [17]	Smart Home Events
Huynh [18]	Daily Routine
Audio Sets	**Target**
DCASE [19]	Acoustic Scene and Events
TIMIT [20]	Phoneme Recognition
Common Voice [21]	Speech Recognition

In context of audio processing, a high number of databases exist that contain recordings of raw audio data. A few popular choices are listed in Table 1.1 and overviews of these sets can be found in [22, 23, 24]. These popular examples are, e.g., the databases of the yearly detection and classification of acoustic scenes and events (DCASE) challenges [19, 25], which includes recordings of acoustic scenes, e.g., office or bus, and events, e.g., speech or alert. The TIMIT acoustic-phonetic database contains thousands of sentences from hundreds of English speakers with different dialects, which are annotated at a phoneme level [20, 26]. All these data sets are not applicable for the daily routine recognition, because they only contain recordings of specific audio examples and does not account for the broad spectrum of life. In particular, day-long raw audio recordings are not feasible due to privacy issues and resulting high amount of data.

In audiological research on hearing aid usages and evaluation, the ecological momentary assessment (EMA) performs long-term monitoring of subjects with objective data and subjective in field questionnaires to avoid the memory bias of retrospective evaluations [27, 28]. Thereby, the objective data are measured every minute and contain the frequency of acoustic class decisions, the mean and maximum broadband level. The hearing aid data was transmitted via Bluetooth to a connected smartphone. Thereby, the real life of 20 experienced HA wearers were recorded for about 3 weeks per participant.

For the daily routine recognition, the Huynh set contains the real life of the author, Huynh, in an open setting by two body-worn ACC sensors [15, 18]. They focused on the daily routine during office working days characterized by a very repetitive structure [9]. To the best of our knowledge, no other suitable public-available data set exists, that contains the real life with head worn sensors. Thus, we identified a literature gap, namely the recording of data sets on high-level activities, to perform further research on model generalization across multiple subjects.

Therefore, one of our main contributions is the time-consuming **creation of two real world data sets** while fulfilling a few important requirements. First of all, since we are dealing with hearing aids, the sensor location is fixed to the ear position on the head, which is not the case for the Huynh data set. In addition, the existing previous studies did not use rich audio features over the long-term periods in this high sampling rate as we do. Hence, there was the need to construct the realistic data sets, $\mathbb{D}_{\mathrm{T}}$ with one person and $\mathbb{D}_7$ with multiple subjects [9, 10]. The

set $\mathbb{D}_{\mathrm{T}}$ is recorded for the feasibility experiments to show which situations are distinguishable and analyze the impact of ACC and audio features. To compare the results of previous $\mathbb{D}_{\mathrm{H}}$ set, we use similar generic activity labels. With the help of data set $\mathbb{D}_7$, we assess the model generalization abilities across subjects for high-level activities. Thereby, we analyze the detection performances in a personalized and person-independent training while using audiological relevant intention-based hearing routine annotations. Both large data sets are recorded in unconstrained environments and contain a broad spectrum of real-life activities. The explanation of the two recorded sets and the Huynh data is given in chapter 2. Thereby, we bridged as one of our main contributions the mentioned literature gap to perform several further investigations introduced in the following sections.

1.5.2 Grouping and Visualizing the Daily Routine Data by Unsupervised Learning

In data clustering, several approaches exist with different concepts to measure the similarity of feature points, for example, based on a distance metric or a density measure [29]. Using a public study data set on accelerometer from smartphones and smartwatches to detect physical activities [30], three clustering algorithms, hierarchical clustering (AGNES), k-means, and density-based spatial clustering of applications with noise (DBSCAN), were tested [31]. The AGNES approach showed to produce the best grouping based on a clustering criterion. In the work of [32], the authors monitored the predicted activity classes in terms of the frequency of changes on a daily basis to judge if the behavior of a certain day is different to others. To do so, they applied k-means clustering on the aggregated output of neural network classifier. Thus, a different daily routine behavior of a person was found on the public van Kasteren smart home data set. To cluster the daily routine data of two accelerometers, Huynh applied an unsupervised topic model (TM) from text processing and generated the words by vector quantization with k-means clustering or classified low-level activities [18]. They found that the topic probabilities correlate with the routine activities.

To gain an understanding of the data set characteristics, visualization techniques are applied [33]. The chosen technique depends on the data attributes, e.g., number of dimensions, temporal or numeric data [34]. A few non-exhaustive examples are line graphs, scatter plots, circular or parallel coordinate displays [35].

Furthermore, dimensionality reduction (DR) techniques can visualize high-dimensional sensor data and a comparative overview can be found in [36]. These methods apply linear or non-linear transformations to reduce the data dimensions while showing the inherent structure. In particular, the t-distributed stochastic neighbor embedding (t-SNE) has shown that it finds good visualizations of high-dimensional data [37].

In our grouping and visualization experiments in chapter 3 on the data sets $\mathbb{D}_{\mathrm{T}}$ and $\mathbb{D}_{\mathrm{H}}$, the following contributions are achieved and partially published in [9]. We demonstrated on the $\mathbb{D}_{\mathrm{T}}$ set that **visualization** plots of ACC and audio features or the activity-loudness map already show **distinguished routine behavior over time**. Additionally, applying dimensionality reduction techniques for visualization purposes on both sets, the evaluation demonstrated that **t-SNE** finds a very **mean-**

ingful embedding of the high-dimensional data. Using the hierarchical clustering method on the t-SNE projection on the $\mathbb{D}_{\mathrm{T}}$ data, the found clusters, working, listening, and talking, form a continuous manifold with similar labels confirming the **manifold assumption**. In addition, the verifying the **clustering assumption**, that, e.g., sport or transport group form own clusters in the feature space. Furthermore, we also tested our visualization approaches and further techniques on the EMA set of [28]. Due to the low sampling rate and low dimensionality of frequency class counts and acoustic levels, the data only showed very obvious strong routine patterns, such as using the car every day on the same time slot or having a high noise level during lunch breaks in a canteen environment.

1.5.3 Labeling the Daily Routine Data by Semi-Supervised Learning

In semi-supervised learning, a model is trained on a small subset of labeled data and this knowledge is transferred to the remaining larger set of unlabeled points. The visual interactive labeling (VIL) [38] combines a state-of-the-art dimensionality reduction technique with the human perception. In a nutshell, the algorithm works as follows: the user selects objects in low-dimensional representation and labels them. Afterwards, the classifier is trained on the manually chosen data points and predicts the remaining objects. This procedure is repeated until the result is visually satisfying. In contrast to active learning, where an algorithm selects the objects, which best optimizes the learning model and asks the user to label them [38, 39]. Both methods work well for annotating high-level activities and daily routines with time diaries. Without too much labeling effort, high-quality annotations are generated by a coarse collection of known activities and routines [40].

In the context of semi-supervised audio processing in [41], events are detected based on weakly labeled data. The problem is formulated as a multiple instance learning, where for each recording, there is only knowledge about the presence of events and not about the exact start and end time. Thus, for each audio frame there is the weak knowledge of a possible presence of one event, but also the strong knowledge of the absence for other events, which are not in a recording. With the help of a support vector machine (SVM) or a neural network, an event detector is built on a Gaussian mixture representation of Mel Frequency Cepstrum Coefficient features. The approach worked well for the detection and temporal localization of specific events like the voices of children, but others, such as laughing, were more problematic to be detected.

Labeling the daily routine data in chapter 4 on the sets $\mathbb{D}_{\mathrm{T}}$ and $\mathbb{D}_{\mathrm{H}}$, the following contributions are achieved and published in [9]. One of our contributions demonstrates that our extension of the VIL method works well for **consistent data annotations** in context of daily routine recognition. It offers the advantages such as spotting of short routine events or a better handling of time-offsets with a coarse time diary. Additionally, the extended VIL approach is highly capable for **data exploration** purposes. The VIL method is validated on the public Huynh data set.

1.5.4 Features for Daily Routine Recognition

The field of human activity recognition has been intensively investigated by focusing on low-level activities. These studies showed good detection rates by finding suitable features for the head and body orientation [42], locomotion [43, 44], conversational gestures [45], and transportation modalities [46]. Thereby, researchers applied statistical, time, or frequency-based features. In audio research, lots of work was spent on good detection features for speech, own voice, and noise by their characteristics [47, 48]. The acoustic features, in particular, are very rich for detecting sound events or characteristic acoustic scenes like certain environments [49], activities of daily living [50], conversations [51], or transportation modalities [52]. This effectively complements the analysis of ACC patterns to differentiate, for example, seated activities as office work vs. having a conversation [53]. The fusion of the HAR and audio fields was done in a few studies for mostly short-term activities such as in a workshop [54, 55]. We build on these results by applying the most suitable features for our DRR use case.

To find an optimal feature representation for the DRR explained in sections 2.4, 3.1, and 5.1, the following contributions are achieved and published in [9]. Based on the set $\mathbb{D}_{\mathrm{T}}$, we analyzed the impact of our features for the classification tasks. That is why, the DRR is performed on the three input data sets: acceleration, audio, and ACC plus audio. With only the ACC data, only for some routines a good result is achieved, but others need to be improved by a better representation. One of our contribution enriches situational details and confirms that our **audio features** are **very informative** and improve the performance of routine classification in comparison to only applying ACC features [9]. In addition, we demonstrated that our selected efficient feature representation is beneficial to **differentiate various daily routine situations and environments**. The applied audio features describe various acoustic properties well, such as music, speech, wind, own voice, or sound level. The acceleration features give information about, for example, the motion strength, orientation, or periodic movements. Furthermore, we showed in chapter 6 that our statistical feature representation is robust against the missing feature problem, since due to the Bluetooth data transmission the features are affected by a varying number of samples.

1.5.5 Classifying the Daily Routine Data by Supervised Learning

In context of the periodic daily routine recognition, only a limited number of authors worked on these composition activities of low-level primitives on the Huynh data set. Blanke first spotted low-level activities on the Huynh set, and then build up a co-occurrence statistic of them within a sliding window [56]. Finally, a joint boosting approach optimizes the statistic and returns the posterior probabilities for each activity routine. In addition, topic models were applied to recognize these high-level activities based on clustered acceleration data in [57, 58, 59]. Huynh used the maximum correlation between the routine class occurrences and the TM probabilities to recognize the daily routine activities. Seiter and White continued the work on TM by analyzing and optimizing the robustness and parameters of the model plus the detection of routine changes. In [60], the parametric work on

TM was extended by a non-parametric formulation to avoid manually selecting the appropriate number of low-level clusters and needed topics.

Further investigations have been carried out on semi-supervised and supervised approaches to reduce annotation effort and test the recognition performances [40]. All this research has been applied on the public Huynh data set $\mathbb{D}_{\mathrm{H}}$, which contains the acceleration data of the author Huynh over seven working days [18].

For the recognition of low-level activities, lots of experiments with different classifiers, such as decision trees or neural networks, are performed for activity primitives mostly [43]. We continued these benchmark evaluations for the daily routine, which is expected to be more challenging due to the higher abstraction level that generates more variability. In addition, it was demonstrated that for low-level activities the person-dependent model outperforms the independent one in various tasks [43]. Thus, in this dissertation we address the question if this also holds for high-level activities.

For the offline recognition on high-level activities in chapter 5, the following contributions are achieved on our data set $\mathbb{D}_7$ of seven people featuring ACC and additional audio data and published in [10]. We confirm that the **personalized model is superior** to person-independent classifier. We demonstrated that the leave-one-fold-out cross-validation returns over-optimistic results due to the temporal correlation of neighboring samples in different folds. We further showed that the **best classifiers, multi-layer perceptron (MLP) and random forest (RF),** yielded the significantly best F-measure performance. In addition, we also processed the Huynh data set with our supervised scheme and outperformed the topic model approach in chapter 4 [9].

1.5.6 Adapting the Classification Models by Online Learning

In the previous sections, the classification models were only once trained and stationary. To better address temporal changes in the data distribution, adaptive models are also considered. The reason is that the long-term behavior of routine activities might change over time and each subject has a different composition of routine events. Therefore, the online model personalization is assessed how well it follows the possible non-stationary behavior. Thereby, the classifiers are updated by adaptation of model parameters, ensemble methods, or incremental updates [61, 62]. A review on incremental learners stated a good tradeoff between the computational efficiency and performance by the linear support vector machine with stochastic gradient descent updates, Gaussian Naïve Bayes, and Online Random Forest (ORF) [63]. In addition, the popular neural networks are incremental learners by performing forward and backward passes on data chunks. Thus, we test all these algorithms and use a as small considered multi-layer perceptron (MLP) network to keep the computational demands still feasible for a HA. The Gaussian mixture model (GMM) was often used in other audio or hearing aid studies for classification due to its computational efficiency. That is why, we also apply it for comparison reasons [49, 64].

One study personalized the HAR model on inertial sensor data without a user interruption [65]. This is achieved by pretraining a user-independent model and performing incremental online updates with the own model predictions on unseen

data. They used the Learn++ ensemble method, which adapts its model based on suitable sized data chunks and tested three types of base classifiers, for example, classification trees. However, the user-independent model must be accurate enough that model personalization can self-improve the recognition accuracy and we test if this also holds for our application. In [66], the HAR model was personalized using an ORF. They updated the ensemble classifier to the new incoming information by adding and deleting classification trees. Both personalization studies worked on low-level activities with small data sets.

We combine these update strategies in our own ensemble approach and cross-compare to existing incremental algorithms. Therefore, we intensively investigate the capabilities of these models to improve with their own predictions or true labels. Afterwards, the online evaluation assesses the performance either with the interleaved test-then-train, the so-called prequential, or holdout evaluation [67]. To compare with the offline baselines, we choose the holdout evaluation.

In our online simulation in section 6.1, the following contributions are achieved on our data set $\mathbb{D}_7$ and published in [10]. The random forest strongly enhanced the F- and accuracy rates using the true and predicted labels compared to the baseline of the initially fitted model. Thus, the **RF classifier** can **self-improve** its model over time and the improvement was significant. Other classifiers only enhanced either the minority or majority class detection.

1.5.7 Modeling the Temporal Daily Routine Transitions by Sequence Learning

To further improve DRR, we model the sequence behavior of the daily routine, which consists of recurring situations and environments. This repetitive time behavior is described by sequence approaches [68]. The classical method is the hidden Markov model (HMM), where a model of each activity state generates the observed data and the transitions between them only depend on the previous state (Markov assumption) [69]. In the domain of routine activities, this assumption does not hold, since the neighboring samples have a correlation that can exhibit longer periods, i.e., from minutes to hours. In contrast to gesture recognition or low-level activities, where these primitive movements have a correlation that can last for a few seconds. For example, an HMM can decode from posture sequences the interest of a child performing tasks [70]. A static posture classifier returns probability scores for the classes, such as lean forward or backward, and the HMM deciphers the posture sequence to one of four interest levels. Furthermore, in assembly tasks, a Gaussian mixture model fitted on ACC data was the input for an HMM to model the transitions between different working steps [55]. Additionally, a second classifier trained on audio features was fused to the GMM-HMM for an optimized decision-making and the audio properties showed to be beneficial. For DRR on the Huynh data set, the GMM-HMM was applied to recognize the daily routine and was inferior to the TM on a long observation window of a half-hour shifted by 5 minutes [18]. Thus, the GMM is often used as a generative observation model [64], but discriminative models can also be applied and demonstrated a better performance [69]. We cross-compared the GMM with our well-performing MLP and RF of the offline and online classification.

Further methods are recurrent neural networks, where we evaluate the performance of a long short-term memory (LSTM) network, since it demonstrated in lots of activity studies a good outcome [71]. For example, the activities of daily living or gestures, such as household tasks, physical exercises, opening a door, or gait parameters, are accurately detected from sensor readings like acceleration or angular velocity data [72]. In particular, the LSTM net favors learning of long-term relationship in data with a natural ordering, which is limited for an HMM due to the Markov assumption [69].

In sequence learning experiments in section 6.2, the following contributions are achieved and published in [11]. We demonstrated that the **multi-layer perceptron and random forest as an observation model** for the **hidden Markov model** achieved the **best F-measure performance** on our set $\mathbb{D}_7$ and the Huynh set. Thereby, the MLP has the strongest F-measure improvement on both sets by adding the HMM. The long short-term memory network has a worse F-measure performance on both sets. The segment error analysis discovers for sequence learners the strong enhancement of temporal prediction stability.

1.6 Summarizing the Publications

The following publications have been written during the time of doctoral candidacy:

INTERNATIONALLY REFEREED CONFERENCE PAPER AND JOURNAL ARTICLES:

- T. Kuebert, H. Puder, and H. Koeppl, "Daily routine recognition with visual interactive labeling by fusing acceleration and audio signals," in *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 1–6, Dec. 2019
- T. Kuebert, H. Puder, and H. Koeppl, "Daily routine recognition for hearing aid personalization," *SN Computer Science*, vol. 2, pp. 1–12, Mar. 2021
- T. Kuebert, H. Puder, and H. Koeppl, "Improving daily routine recognition in hearing aids using sequence learning," *IEEE Access*, vol. 9, pp. 93237–93247, June 2021

INTERNATIONAL PATENT AND PATENT APPLICATIONS:

- T. Kuebert and T. Wurzbacher, "Method for operating a hearing aid, and hearing aid," United States Patent Application US20210051420A1, Feb. 2021
- T. Wurzbacher, T. Kuebert, and D. Mauler, "Method for operating a hearing device and hearing device," United States Patent US010959028B2, Mar. 2021
- T. Kuebert and S. Aschoff, "Method for the environment-dependent operation of a hearing system and hearing system," United States Patent Application US20210176572A1, June 2021

1.7 Structuring the Thesis

The structure of the thesis is illustrated in Fig. 1.2 and explained in the following:

- **Chapter 2** describes the two own recorded real-world sets of realistic hearing aid data. The set $\mathbb{D}_{\mathrm{T}}$ with generic activity classes is utilized for the feasibility experiments, and the set $\mathbb{D}_{7}$ with the intention-based hearing routine labels builds the ground for the model generalization investigations. The public Huynh set $\mathbb{D}_{\mathrm{H}}$ serves as a reference for comparisons and evaluation purposes. The hearing aid features, audio and acceleration, are explained plus the feature notation is introduced.

- **Chapter 3** analyzes if the routine data groups into relevant situations and environments by applying clustering methods. To do so, an efficient feature representation is extracted on the Huynh and $\mathbb{D}_{\mathrm{T}}$ set. Various visualization techniques are used to show the temporal structure of the routine data. Thereby, several dimensionality reduction techniques are evaluated.

- **Chapter 4** annotates the data set $\mathbb{D}_{\mathrm{T}}$ to recognize the daily routine by semi-supervised learning. The extended visual interactive labeling (VIL) method is explained and applied. Based on these activity annotations, the routine is recognized and the effect of different input data modalities, audio and acceleration, is analyzed. The Huynh set serves as an independent evaluation set for the VIL method.

- **Chapter 5** performs the daily routine recognition based on various classification techniques. For this purpose, an improved feature representation for the classification is extracted and automatically selected. Various supervised techniques are introduced, and the model generalization abilities are assessed in a personalized and person-independent model training.

- **Chapter 6** improves the routine recognition by online classification approaches and sequence modeling techniques. Thereby, an online model is initially trained in a person-independent manner and fine-tuned on the personal data stream. Sequence models improve the classification results by exploiting the temporal routine stability.

- **Chapter 7** summarizes the achievements of this thesis and gives an outlook for further work.

- **Chapter A** is the annex, introduces the study protocol for the data set $\mathbb{D}_{7}$, and explains the recording procedure. Furthermore, the audiological optimization of hearing parameters based on routine classes is discussed and a solution is proposed. Additionally, means to integrate the routine classification with the acoustic classifier of a hearing aid are explained.

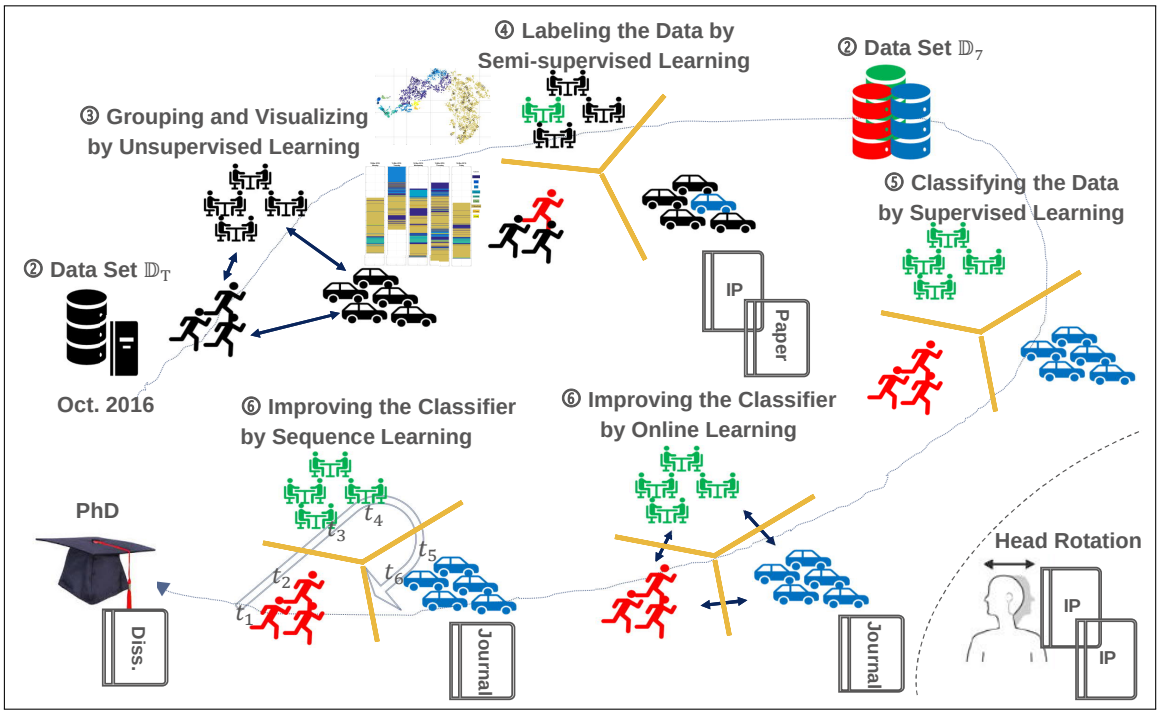


Figure 1.2: The overview of the dissertation with the corresponding chapter numbers in the circles illustrates the machine learning journey sorted by content towards the PhD starting in October 2016. We display the different learning principles in a two-dimensional feature space based on the example of three prototypical situations: a sport exercise, a social conversation, and a car ride. Thereby, the two own recorded real-world data sets, $\mathbb{D}_T$ with a coarse time diary of activity classes and $\mathbb{D}_7$ with the color-encoded intention-based user labels, build the grounds for our comprehensive experiments with acceleration and audio data. The initial investigations are performed on the $\mathbb{D}_T$ set and analyzes with the help of clustering and visualization techniques how well the routine data groups and which situations are distinct in the feature space. The labeling approaches annotates the set $\mathbb{D}_T$ based on the sparse time diary knowledge by training a semi-supervised algorithm, where the gold lines denote the decision boundaries. After the labeling, the routine activities are recognized, and the effect of different input data variants is analyzed. Based on the annotated data set $\mathbb{D}_7$ with seven subjects, supervised techniques are evaluated with a fixed decision boundary between the classes. For improving classification, the online methods adapt the boundaries to personalize the classifiers, while the sequence algorithms model the temporal relationships between the samples. The corresponding conference and journal publications plus patents (intellectual property, IP) introduced in section 1.6 are placed by the covered topics. The initial research work following the master thesis to detect head rotations leaded to two IPs and was beneficial to find suitable motion features. Two pictures, PhD hut and rotating head, are taken from [76, 77].

Chapter 2

Daily Routine Data Sets and Labels

In this chapter, we introduce the characteristics of the data sets and labels on which we perform the routine analysis. Our goal is to analyze and recognize recurring daily routine situations. Therefore, we perform realistic and unconstrained recordings with hearing aids. To do so, we discuss and evaluate hearing needs of common relevant acoustic situations and environments in section 2.1. This allows to derive the used class annotations, which are meaningful for hearing-impaired people in section 2.2. Afterwards, the data sets and the recording procedure is explained in section 2.3. A first recorded set with realistic hearing aid data and one subject serves for the feasibility experiments and has offline annotations of a time diary. Later, a second real-world data set of multiple subjects with intention-based hearing routine labels is recorded to analyze the model generalization abilities. A further public-available data set is explained for reproducibility and comparison reasons. The features of the two real-life data sets are presented in section 2.4. The material introduced in this chapter is partially taken from our publications [9, 10, 11].

2.1 Common Hearing Situations and Labels for Classification System

To derive relevant class annotations for our recordings, we discuss common hearing situations and consider the existing concepts to categorize and annotate hearing scenes. This allows to record representative data of the recurring hearing situations and environments with relevant class annotations. The first concept is the auditory context that describes and analyses the hearing task based on the acoustic environment, social interaction, activity location and content [78]. The acoustic environment is characterized by sound properties, such as the distance to a target speaker, the strength of reverberation, the noise level or type. The social interaction determines the importance of a situation, for example, to understand someone in group conversation or in a dialog. The activity location influences how difficult it is to understand someone, e.g., in a train station or at home. Thus, all these characteristics influence each other and determine how challenging the hearing task is. For example, a conversation between two friends in calm cafe can be an easy hearing task, whereas the same conversation in crowded restaurant can be difficult. The

analysis of the auditory context states that the elderly people spent most of time at home. However, the most important situations with the strongest hearing needs usually are in unfamiliar locations, which makes them more difficult. The content can be having a conversation, listening to live events or media, talking on the phone, or passive listening events. The distribution of these activities significantly varies across users, which creates the need for a personalized treatment. Mostly, the time is spent in low-noise environments. According to [79], elderly people tend to have quieter auditory lifestyles, since they have less active social lifestyles, which results in fewer hearing demands.

Further categorizations for everyday sound environments are defined by the common sound scenarios [80]. They are experienced on a daily basis by most people and are defined by their sound and environmental properties. They are grouped based on the hearing intention into three main characteristics:

- having a speech communication with at least 2 people in live or through a device,
- focusing on listening to live sounds and through a media device, and
- non-specific situations while monitoring the surroundings (e.g., during running) or passive listening (e.g., unconsciously during a train ride).

For each of the three scenarios, the context is defined by the hearing intention and task, i.e., the intention determines the hearing task in a specific scenario. For example, during train ride you can actively participate in a conversation or just passively listening to it. Thus, the hearing task in an active discussion would be to understand every phrase of the participants and filter out the surrounding noise sounds, whereas in a passive mode you follow up more on trigger words and ignore most of the content. Therefore, the hearing intention and task are two very important parameters for an optimal hearing outcome. The data analysis from real hearing situations assesses the importance and frequency of certain environments and situations [27]. Thus, elderly people perform, in general, many home activities and only a few job, transportation, or social activities. The inter-individual differences in duration or frequency across subjects require an individual analysis and creates the need for a personalized assessment.

These categorizations of hearing scenes build the ground for various hearing systems, but the ideal one is the full-functioning human brain, which effectively adapts to various acoustic situations. It is described by the theory of auditory scene analysis [81]. The individual acoustic sources, e.g., voices or music melodies, sum up to a sound wave, which is received by the ear. The human brain analyzes these incoming signals by applying heuristic processes. These can decode various acoustic properties, e.g., the perceived pitch, loudness, timbre, and spatial perception. Therefore, with various acoustic situations can be effectively dealt. Most importantly, the brain knows the current intention of a person and can optimally adapt to the scene characteristics, since it knows which properties matter the most and can emphasize these components. Technical systems try to mimic these adaptation processes by classification systems [82, 5, 28]. These translate the hearing problem into the recognition of prototypic acoustic environments. Common class choices are:

- speech presence (conversations, lectures, calls, television),

Table 2.1: List of intention-based routine classes and corresponding exemplary activities in the data set $\mathbb{D}_7$.

Routine Classes	Activities
Transportation	Commuting, train, car, bus, plane, location change
Physical (Activity)	Exercises, sport, manual work
Basics	Hygiene, dressing, resting, eating, preparing food, housekeeping, office work
Social (Interaction)	Family, friends, conversations, partying, play music, singing, call
(Focused) Listening	Music, cinema, theater, concert, lecture, TV, media

- speech in noise (cocktail party situation, announcements during train rides, a conversation in a car),
- silence,
- noise (construction work),
- music, and
- car.

For these classes, predefined processing schemes for the incoming sounds are defined. Hearing aids switch between these classes, the so-called hearing programs, which should fit best to these situations. These programs are trained in a person-independent way, i.e., they follow one fit for all strategy. The switching frequency between programs was rated quite differently by hearing aid wearers, i.e., individual settings are preferred [82]. Unfortunately, the pure acoustic classification has an ambiguity problem in scenarios, where the intended hearing wish is not clear by only considering sound properties, since multiple hearing wishes are possible in some acoustic environments [2]. Thus, further information, additional sensors, or a personalized classification scheme is needed to solve this intention problem.

2.2 Intentional Daily Routine Labels

To solve the intentional decoding problem of hearing situations and environments, we propose a personalized classification, and therefore the user selects the routine class labels based on his intention. Furthermore, the routine labels should address the most important situations and hearing tasks discussed in section 2.1. That is why, the proposed routine classes have different hearing needs and are listed in Table 2.1. The hearing task depends on the class and has different targets depending on the acoustic environment, situational intention and importance. Starting at the top of the list, the transportation routine accounts for all modalities, such as car or bus, which go from A to B. The hearing task would be to passively monitor the surroundings. While the physical activity stands for high-intensity routines, such as

sport exercises or manual work, where the task could be to filter out construction noise and actively monitor the surroundings. On the contrary, the basics group includes low-intensity activities and is inspired by the activities of daily living (ADL) [83]. The ADL concept represents the fundamental functions of living, such as eating or hygiene. Further activities, such as office work or reading a newspaper, are included as well and have less focus on the acoustic environment.

The next two routine classes are influenced by the so-called common sound scenarios and are the most difficult situations for the hearing-impaired people [80]. The social (interaction) routine is the most crucial to participate in life during conversations in various environments. Here, in particular, the hearing task is to fully understand every word and emphasis to optimally respond in a chat. That is why, the social routine is the most demanding and important hearing situation. Likewise, the (focused) listening routine is another fundamental function for the hearing to receive information from media or joy from music. Here as well, it is important to follow the activity, but since no response is needed less focus is necessary than in a social situation. These two hearing functions are sometimes determined by the intention of the wearer in the situation. That is why, the user should select the intended dominant routine, i.e., in a conversation during a car ride, the dominant routine would be social. Hence, the classes are not mutually exclusive, which may be a possible source of confusion for the classifier and may result in a lower recognition rate. But we assume, that the situational intention changes the motion behavior allowing us their detection. For example, in a social conversation we assume more head movements than in acoustically similar listening situation, which allows to differentiate both situations.

The introduced classes correspond to different hearing needs, which require specific signal processing settings. A few non-exhaustive examples are mentioned to gain a better understanding of the routine class goals. In a listening or social situation, it is often required to focus on a target speaker, where directional hearing is beneficial. Whereas, in basics, transportation, or physical class an omni-directional setting helps to keep the situation awareness and monitor if someone approaches the HA user. In a car transport scene, a typical low-frequent noise is present, that creates the need for noise reduction measures.

These five classes are intention-based prototypes, but individual classes for each user are also possible. In this work, we use the predefined generic class profiles to make our study feasible for evaluation purposes and comparisons. This allows to assess the model generalization across subjects, which would not be possible with individual user profiles.

2.3 Data Sets and Recording Procedure

In this section, we introduce our two large real-world HA data sets and one public set. The first set $\mathbb{D}_{\mathrm{T}}$ contains the real life of Thomas with a feature-recording app plus generic time diary annotations. It serves for the feasibility experiments to show the proof of concept. The second set $\mathbb{D}_7$ includes the realistic recordings of seven subjects with an extended mobile application for further features and intention-based user labels. It is utilized for the model generalization experiments across subjects. The public Huynh set $\mathbb{D}_{\mathrm{H}}$ is applied for comparison and evaluation reasons. It contains the real life of Huynh with four generic routine classes.

Table 2.2: Summary of recorded feature data on the used sets with the sampling rate and input data source. The hearing aid features of sets $\mathbb{D}_{\mathrm{T}}$ and $\mathbb{D}_7$ are the same with different sampling rates and explained in section 2.4.

Set	Input	Data	Rate
Thomas $\mathbb{D}_{\mathrm{T}}$	Hearing aid (14D)	Own voice soft and hard activation, temporal level correlation, spectral centroid noise floor, low- and mid-frequency noise floor, stationarity, 4 Hz modulation, onset detection, wind, max. level, acceleration	$\leq$ 16 Hz
	iPhone	Timestamp (generated on data arrival)	$\leq$ 16 Hz
7 Subjects $\mathbb{D}_7$	Hearing aid (11D)	Own voice soft and hard activation, temporal level correlation, spectral centroid noise floor, low- and mid-frequency noise floor, stationarity, 4 Hz modulation, onset detection, wind, max. level	2 Hz
	Hearing aid	Acceleration (two-time steps: 16 Hz), temperature	8 Hz
	iPhone (14D)	Timestamp, acceleration, magnetometer, gyroscope, orientation quaternion, user activity	8 Hz
	User	Label	8 Hz
Huynh $\mathbb{D}_{\mathrm{H}}$	Pocket sensor	Acceleration, timestamp	100 Hz
	Wrist sensor	Acceleration, timestamp	100 Hz

2.3.1 Data Set of Thomas $\mathbb{D}_{\mathrm{T}}$

The data set $\mathbb{D}_{\mathrm{T}}$ is recorded to demonstrate the feasibility of our intended daily routine approach with only one subject. This means we analyze if the selected features are informative for the routine behavior and discriminative to recognize various routine situations and environments. Thus, it is utilized for the grouping and labeling experiments in chapters 3 and 4. The set consists of Thomas' daily routine, which has been recorded for $N = 4016$ minutes by two Signia Nx hearing aids. The devices are worn behind the ears and are shown in Fig. 2.9. The receivers are positioned and fixed in the ear canals based on suitable sized domes. Each HA has a battery cell, which needs to be manually changed in regular intervals. As an input for the recordings, each prototype hearing aid contains two microphones plus a three-dimensional acceleration sensor. The raw audio signals are transformed to the frequency domain and the sound features are calculated within the HA, which is explained in section 2.4.1.

The transmission is triggered per Bluetooth request by an iPhone. It results in a continuous stream of precomputed audio features and raw acceleration data. A summary of the recorded features is given in Tab. 2.2. This procedure allows

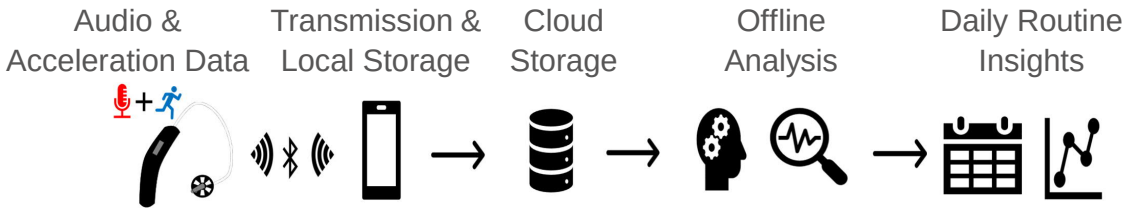


Figure 2.1: The data flow for the data sets $\mathbb{D}_{\mathrm{T}}$ and $\mathbb{D}_7$ from a hearing aid to the daily routine insights. The derived audio features are computed in the HA together with the measured raw acceleration data. Both inputs are transmitted per Bluetooth to the connected mobile phone for the local storage. At the end of a day recording, the stored data is uploaded to the cloud server for a later offline processing in MATLAB and Python. This results in the daily routine insights.

a variable sampling rate up to 16 Hz, depending on the wireless connection and mobile application. Furthermore, the data timestamps are generated by the cell phone introducing a transmission jitter in the timestamps, since the time offset between data generation in the HA and the data storage in the mobile phone is variable. That is why, the recordings of two HAs are not synchronous in time, which is prerequisite for a binaural processing of head rotation features with the acceleration signals [84]. The audio data are the same for both HAs. That is why, we only use the right side for the data set and do not compute binaural acceleration features. An overview of the data flow is illustrated in Fig. 2.1, where stored data of the mobile phone is uploaded to a cloud server to perform the offline routine analysis in MATLAB and Python, which is explained in the following chapters.

Moreover, the evaluation is based on a time diary and offline annotations since the mobile application has no labeling functionality. This labeling procedure does not introduce unnatural interruptions and a priori unknown activities can be handled as well [18]. The annotations are displayed in the time schedule in Fig. 3.7 (a).

The most prominent activity of our data set is (office) work, which mostly consists of sitting at the desk and working on a computer with smaller interruptions such as coffee breaks. Usually, during lunch break the nearby canteen and afterwards a cafe are visited by foot. Both environments have loud babble noise. Within the working day, some meetings plus general conversations are included in the talk routine. Furthermore, the main mode of transport is the bicycle, which is only used for several minutes a day. Only once in the data set approximately one hour, a shared car ride is taken. Typical evening routines are watching TV or going to nearby fitness center. The chosen routines represent the full spectrum of life.

2.3.2 Data Set of Seven Subjects $\mathbb{D}_7$

Our second set serves as a basis for the model generalization analysis to recognize the daily routine across multiple subjects. It is utilized for the offline, online, and sequence classification experiments in chapters 5 and 6. In contrast to the $\mathbb{D}_{\mathrm{T}}$ and $\mathbb{D}_{\mathrm{H}}$ data with 1 subject each, the set $\mathbb{D}_7$ contains the real life of 7 subjects. The 3 females and 4 males have a relatively low mean age of 29.3 ± 8.9 years for HA users. Hence, they are not representative for hearing aid customers, which are mostly in retirement age over 60 years old [3, 78]. Thus, it is expected that the younger people

have a more active social lifestyle with more demanding hearing situations, which makes our task more challenging [79]. The task for the subjects was to record the personal daily routine as long as possible (mean duration per day of 610.1 $\pm$ 166.7 minutes) over a longer period of time (mean number of 14.9 $\pm$ 3.4 days).

During the total length of $N = 63449$ minutes for 104 days, the Signia Nx hearing aids introduced in section 2.3.1 are worn on the ears and have a special firmware to continuously streams the data via Bluetooth services to the mobile phone. Again, as for the set $\mathbb{D}_{\mathrm{T}}$ with the same data flow in Fig. 2.1, we only use the right side for the analysis. An overview of the recorded data is given Tab. 2.2. The precomputed audio and raw acceleration features are ideally sampled at 2 Hz and 16 Hz, but sometimes due to transmission problems, the rate can be lower. The rates are a good compromise between transmission stability and detection performance [43]. The variable rate of the data transmission leads to missing feature samples over time, which can have an influence on the classification performance [85]. Since our features are highly correlated over time from seconds up to minutes, the neighboring samples have similar information, i.e., the negative consequence of losing samples is reduced. Additionally, we design statistical features in section 5.1 that can deal with variable number of feature samples. Thus, our daily routine detection is resilient to the missing feature problem.

Camera or raw audio recordings were considered but are not feasible over a long period of time and would be a privacy issue, especially in public environments. In contrast, our design is less obtrusive enabling the subjects to behave as naturally as possible. The participants were instructed how to use the mobile application and the labels. As a resource, each subject had the study protocol and manual, which is depicted in Fig. A.1 in the annex in section A.1. A miscellaneous (misc) class was used in the case a user did not choose any label at the start of a recording or to indicate something went wrong. The misc instances were then relabeled or excluded from the data analysis.

Furthermore, the data timestamps and user annotations for the evaluation are generated in the iPhone application. Again, we only use the data of one HA due to the asynchronous recordings of both devices. The iPhone detects the following activity classes: stationary, walking, running, driving, cycling, and unknown. These activities can be also detected by the built-in ACC sensor in the HA and iPhone software updates may change the class detection behavior. Thus, we only use the timestamp of the iPhone and discard the rest for further analysis. The iPhone inertial sensor data are not used, since during the recordings the values were frozen due to energy saving mechanisms, which cannot be turned off. The temperature property of the accelerometer did not provide a reliable indicator, since it is relative to environmental temperature depending on the weather or heating. Thus, it was omitted for the further analysis. But it could be utilized for an online zero-g bias tracking, which is temperature-dependent.

The users can report label errors or general problems, e.g., due to forgotten annotations, disconnection of Bluetooth transmission, or time offsets, in the recording app for a later manual correction and shortly summarize their day. A few exemplary user comments are shown in Fig. 2.2, and we discuss them in a bottom-up order. One user also expressed the ironic gratitude for taking part in these long recording experiments, which can be exhausting. Another user summarizes the day and gives details to the recorded situations and occurring problems. The third and fourth

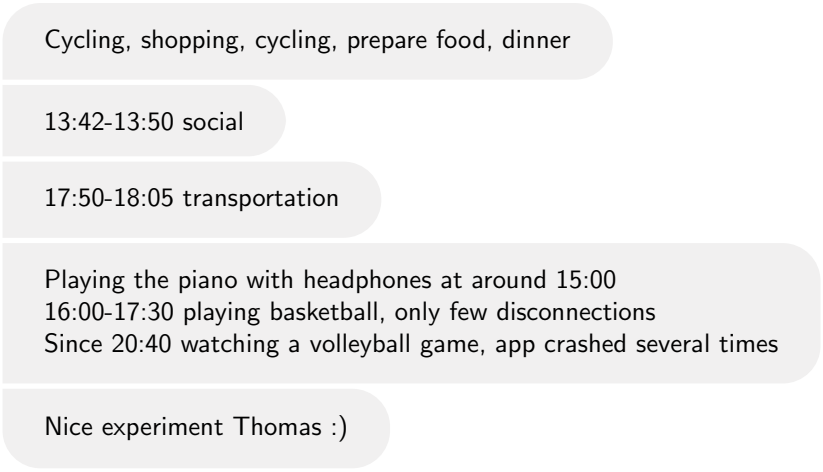


Figure 2.2: Exemplary user comments on our data set $\mathbb{D}_7$.

comment refers to a forgotten transportation and social annotation for the manual correction. The last comment gives an overview of the sequence of recorded events.

In Fig. 2.3, two example days, a workday and a day of weekend, are shown, where the working day has a typical structure for the $\mathbb{D}_7$ set. Before and after the office work, there is a transportation scene for commuting and during the day multiple short conversations happen plus two longer ones during lunch and dinner time. These working days follow a natural ordering, which is beneficial for learning these relationships. At the weekend, the day structure is less ordered between different day examples and more free time activities are performed like in Fig. 2.3 (b), where a longer period of the physical class happens. Besides, people are more socially active during weekends, which is, in particular, the case for young people having more active lifestyle [79]. Additionally, not all classes are active every day, e.g., the physical class on the working day example. This is known as reoccurring concept drift [61].

The overall main activity routine is the office work, which is part of the basics class. The lunch break in the nearby canteen is labeled as social class, has a loud babble background noise, and is visited by foot. Within the working day, some meetings plus general conversations are included in the listening or social routine. Furthermore, the main mode of transport is the bicycle or car for commuting. Typical evening routines are meeting friends as social class, watching TV as listening class, dancing, or going to the fitness center as physical class. Five subjects had the described office work routine containing lots of repetitive situations and environments. Two subjects followed a less recurring schedule and had more free time activities.

Furthermore, the prior class distribution is shown in Fig. 2.4 across the subjects, where we find stronger inter-individual differences. For example, subject 1 has the strongest share of 52.9% in the basics class due to long office work sequences, whereas subject 5 has the smallest proportion of basics with 23.4%. Both users mark the extremes around the mean percentage of 35.8%. The physical class has a small range from 1.8% of subject 3 to 13.3% of subject 6, i.e., the personal activity patterns produce strong differences in the physical class frequency. Similarly, the transportation class spans a small interval from 1.6% of subject 2 to 12.0% of subject 3 depending, e.g., on the personal commuting distances and living environment. The

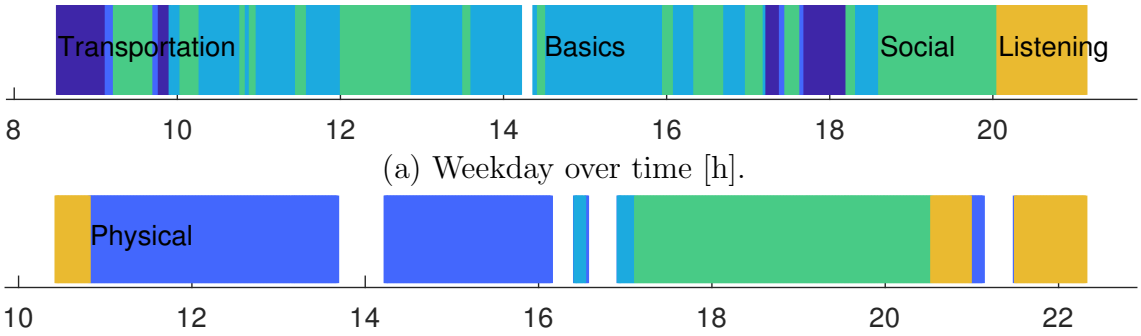
second strongest majority class social also highly varies across the individuals from 27.1% of subject 1 to 51.2% of subject 3 with a mean share of 40.4%. The social and listening classes have conflicting trends, i.e., strong talkers like subject 6 with a social share of 48.0% listen less with only a percentage of 3.6% and vice versa. The listening class spans a range from 3.6% of subject 6 to 18.9% of subject 5. Hence, the different personalities and routines affect the prior class distribution, i.e., some tend to be more talkative and others more a good listener [86]. Thus, the classes are differently imbalanced across subjects.

In Fig. 2.5, we show the event duration analysis of our set $\mathbb{D}_7$ for all subjects together, since the inter-person variability is very small. Most situations have a short duration of a couple minutes and fewer events have a long duration of hours. Thus, the duration probability density function follows an exponentially decaying relationship. In addition, we note a variable duration of class events and a couple of transitions between the classes as shown in Fig. 2.3 for the two example days. In general, all 25 possible transitions between the classes occur, but not every day. The day structure is variable across subjects and days, i.e., weekends have a different routine order. Therefore, we have a complex learning task with a high variability and realistic daily routine situations.

To systematically analyze the interday class prior variability, we plot the distribution of the daily class prior in Fig. 2.6 summarized for all subjects together. The interday class prior variability is person-dependent. Obviously, the two majority classes, basics and social, have the highest median values of 41.8% and 34.8%. They also have the highest daily variability of 22.4% and 18.6% measured by the interquartile range. In contrast to the majority classes, listening only has a variability of 14.1% and the remaining two classes fall below 10%. On some days, a class does not occur at all, but on other days it is the majority class. The social routine is the only classes that occurs on every day. Additionally, the interday class variability has a stronger subject dependency, i.e., the median value per subject and class has a range from 8.5% for transportation up to 33.3% for the basics class. Thus, the strong interday class variability makes the learning task challenging.

2.3.3 Data Set of Huynh $\mathbb{D}_{\mathrm{H}}$

The publicly available Huynh data set $\mathbb{D}_{\mathrm{H}}$ is utilized for reproducibility, evaluation, and comparison [18]. The set contains the real life of the first author, Huynh, in an open setting. The two triaxial ACC sensors were sampling at 100 Hz and were placed at the dominant wrist and in the right pocket as shown in Tab. 2.2. The online annotations were created by a combination of experience sampling, time diary, and camera snapshots. Afterwards, they aligned these annotation sources along with the visualized sensor readings. The most frequent routine class is (office) work and the remaining three activities - commuting, lunch, and dinner - happen only with a single-digit percentage. Unlabeled segments are not considered for the analysis. The temporal structure is visualized in Fig. 2.7 and is representative for all seven working days with marginal changes in duration and start times of single class occurrences. To systematically analyze the interday class variability, we plot the distribution of the daily class prior in Fig. 2.8. Obviously, the majority class, work, has the highest median value of 76.9% and covers the main part of the day. In contrast to the majority class, the lunch routine has a median value of 11.8% and



(a) Weekday over time [h].

(b) A day of weekend over time [h].

Figure 2.3: Two example days, a workday and a day of weekend, on our set $\mathbb{D}_7$.

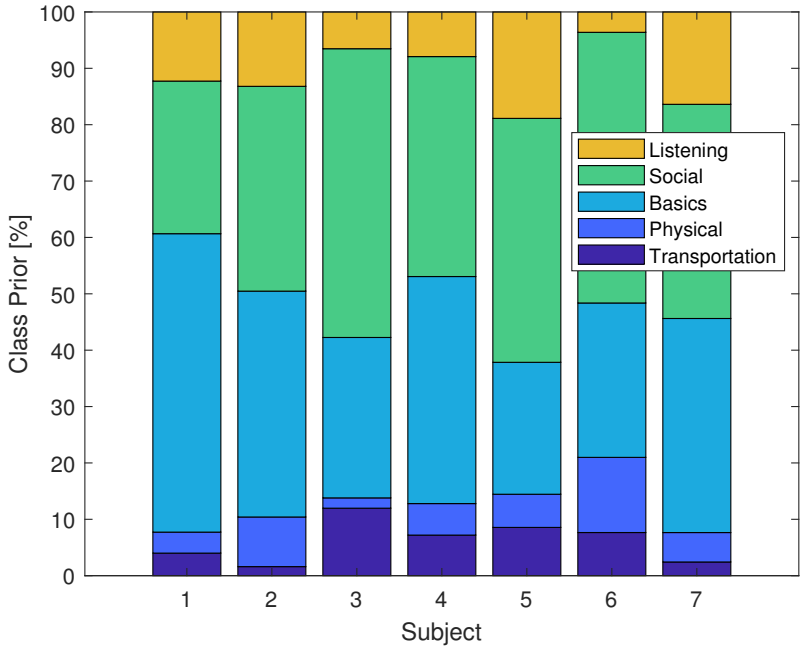


Figure 2.4: Class prior per subject on our data set $\mathbb{D}_7$ shows a strong inter-subject variability.

the two remaining classes, dinner and commuting, fall below 10%. All classes only have a small interday variability, which is lower or equal than 3.8%. Therefore, the interday class variations are very low in comparison to the data set $\mathbb{D}_7$.

The defined classes contain a natural order, i.e., only certain transitions are possible, e.g., commuting to work or vice versa, which reduces the possible complexity of the learning task. The class durations are only variable in a limited amount. Thus, the learning task is simpler for the Huynh set than for our set $\mathbb{D}_7$ and we expect a better detection performance for the Huynh data. In both data sets, short recording breaks occur, since the data are uploaded, device batteries are changed, or transmission links need to be restarted.

To the best of our knowledge, no other suitable public-available data set exists, that contains the real life with head or body worn sensors. In contrast, our set $\mathbb{D}_7$ contains the real life of 7 subjects and is a more realistic with unconstrained environments and activities. Since we are dealing with hearing aids, the sensor location is fixed to the ear position on the head, which is not the case for the Huynh

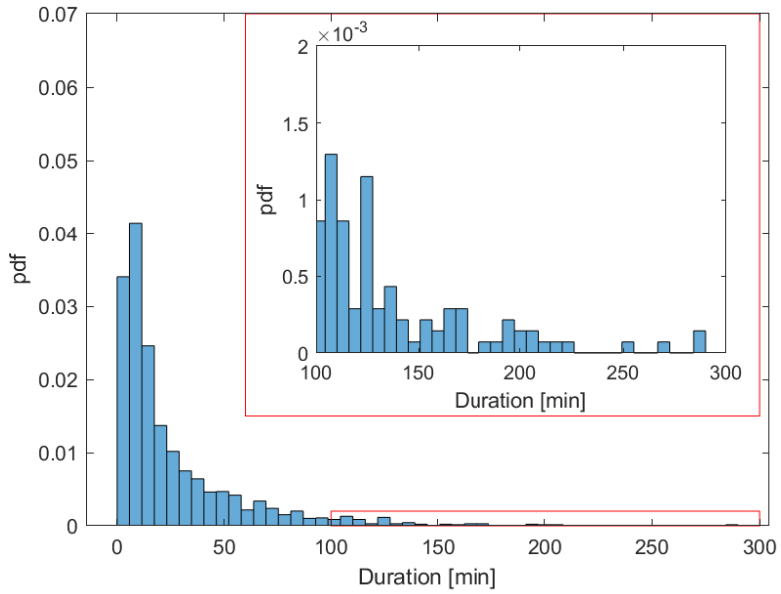


Figure 2.5: Event duration on our data set $\mathbb{D}_7$ with an exponential decaying probability density function (pdf), where the red box is the zoomed-in area.

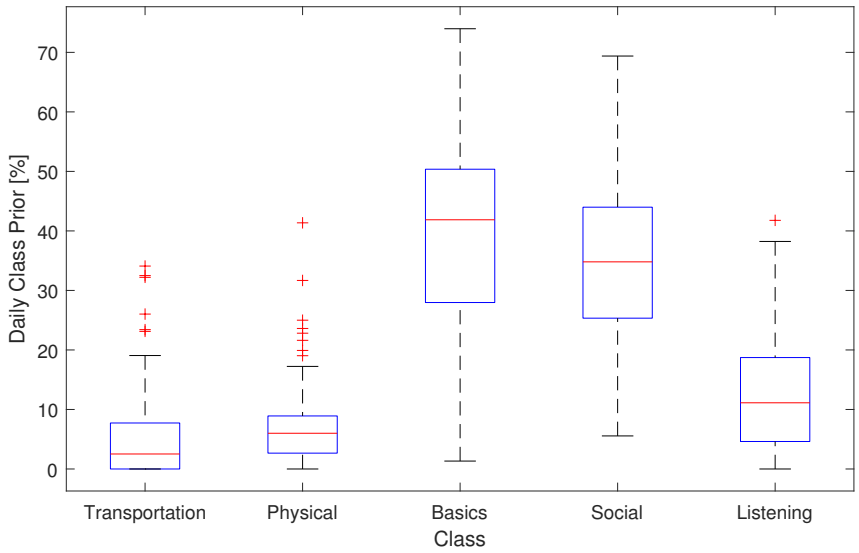


Figure 2.6: The daily class prior of all subjects in the set $\mathbb{D}_7$ shows a strong interday variability.

data set. In addition, the existing previous studies did not use rich audio features as we do. Hence, there was the need to construct the realistic data sets, $\mathbb{D}_{\mathrm{T}}$ with one person for the feasibility experiments and $\mathbb{D}_7$ with multiple subjects to assess the model generalization abilities.

2.4 Description of Hearing Aid Features

In the following, details to the features of our sets, $\mathbb{D}_{\mathrm{T}}$ and $\mathbb{D}_7$, are given. The features are chosen to distinguish the classes by representing the routine behavior and environments well. Their space can be partitioned in two independent inputs:

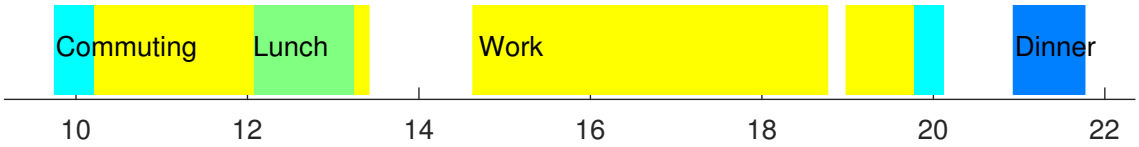


Figure 2.7: Representative example day of the Huynh set over time [h].

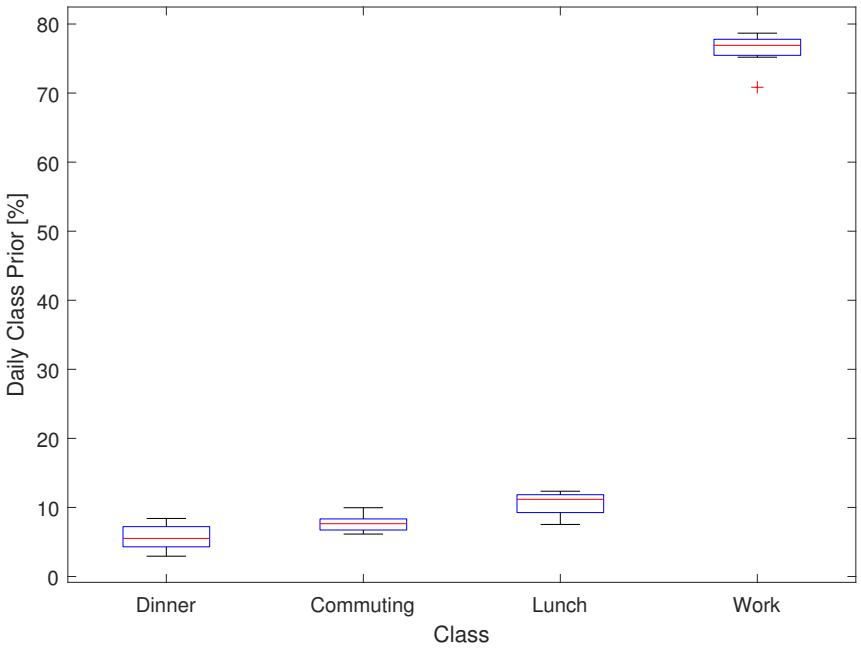


Figure 2.8: The daily class prior in the set $\mathbb{D}_{\mathrm{H}}$ has a small variability for all classes and the work routine is the majority class.

ACC and audio. For the $\mathbb{D}_7$ set, the raw acceleration is measured at a rate of 16 Hz and the precomputed audio features have a rate of 2 Hz, whereas the $\mathbb{D}_{\mathrm{T}}$ set uses up to 16 Hz for both inputs depending on the wireless connection. An example plot of the features is given in Fig. 3.4.

2.4.1 Acceleration Signal Model

A typical state-of-the-art signal model as in [87] is the following one:

$$\mathbf{a}_{\mathrm{mes}} = \mathbf{R}\mathbf{S}_{\mathrm{un}}(\mathbf{a}_{\mathrm{dyn}} - \mathbf{g}_w) + \mathbf{b} + \mathbf{n} \quad \text{in } [g = 9.81\frac{m}{s^2}], \tag{2.1}$$

where the measured ACC $\mathbf{a}_{\mathrm{mes}}$ is expressed in multiples of gravity in the sensor coordinate system and is determined in the interval of $\pm 2g$.

- Rotation matrix $\mathbf{R}$ depends on the attitude of the sensors relative to the head or world coordinate systems and the cross-axes alignment, i.e., the coordinate axes may not be perfectly orthogonal to each other.

- Sensitivity matrix $\mathbf{S}$ has a scaling factor per axis for the analog digital conversion and is represented by a diagonal matrix. In the uncalibrated case, it is denoted by $\mathbf{S}_{\mathrm{un}} = \mathbf{I}$.

- Dynamic ACC $\mathbf{a}_{\text{dyn}}$ depends on the head and body motion, in case of no motion it is zero.

- Gravity $\mathbf{g}_w = [0, 0, -1]^{\text{T}}$ pulls downwards in the world frame and its norm is always one, but it is measured in the opposite direction, which is mathematically equivalent to a reflection on the origin. In the sensor frame, the contribution to the axes $\mathbf{g} = \mathbf{R}\mathbf{g}_w$ depends on the current attitude of the sensor. In the case of free fall, no ACC is measured due to gravity.

- Bias $\mathbf{b}$ is temperature dependent, slowly changes over time, and is also known as zero-g-offset.

- Noise $\mathbf{n}$ is additive white measurement noise with a zero-mean Gaussian distribution.

The part of interest is the dynamic ACC $\mathbf{a}_{\text{dyn}}$ for a rigid body as described in [87]. It assumes the body is not deformable and the HA is not moving on the ears relative to the reference coordinate systems, i.e., $\mathbf{r}$ is fixed. The dynamic ACC is a combination of the rotational and the linear translating ACC:

$$\mathbf{a}_{dyn} = \mathbf{a}_{lin} + \overbrace{\underbrace{\boldsymbol{\omega}}_{\text{angular velocity}} \times \boldsymbol{\omega} \times \mathbf{r}}^{\text{radial ACC } \mathbf{a}_{\text{R}}} + \overbrace{\underbrace{\boldsymbol{\alpha}}_{\text{angular ACC}} \times \mathbf{r}}^{\text{tangential ACC } \mathbf{a}_{\text{T}}} . \tag{2.2}$$

Considering the measurement model of the acceleration sensor in Equation (2.1), the need for a sensor calibration is evident. In literature, two main ways to address the problem are, that the absolute device orientation $\mathbf{R}$ relative to gravity $\mathbf{g}_w$ is assumed to be either known or unknown. Therefore, the task is to measure the gravity vector $\mathbf{g}$, when the device is not moving, i.e., $\mathbf{a}_{\text{dyn}} = \mathbf{0}$. The vector $\mathbf{g}$ is only dependent on the sensor orientation $\mathbf{R}$ and is expressed in the sensor coordinate system. The noise term $\mathbf{n}$ is averaged out over a longer observation window, where the acceleration sensor is in a static position. The calibrated triaxial vector $\mathbf{a}_{\text{cal}}$ is given by

$$\mathbf{a}_{\text{cal}} = \mathbf{R}\mathbf{S}(\mathbf{a}_{\text{mes}} - \mathbf{b}), \tag{2.3}$$

The first approach of [88] uses the orientation information to compute the ideal value for gravity in the body frame and compares the value to actual measured $\mathbf{g}$. A measurement of six distinct attitudes is needed, where each sensor axis is positively and negatively aligned with gravity. Based on this method, the sensitivity, bias, and cross-axes alignment can be determined. The latter way, the so-called ellipsoid calibration, bears on the property, that the norm of the signal should be one:

$$1 = {\mathbf{a}_{\text{cal}}}^{\text{T}}\mathbf{a}_{\text{cal}} = {\mathbf{a}_{\text{mes}}}^{\text{T}}\mathbf{S}^2\mathbf{a}_{\text{mes}} + \mathbf{b}^{\text{T}}\mathbf{S}^2\mathbf{b} - 2{\mathbf{a}_{\text{mes}}}^{\text{T}}\mathbf{S}^2\mathbf{b}, \tag{2.4}$$

which uses the orthonormal property of the rotation matrix $\mathbf{R}\mathbf{R}^{\text{T}} = \mathbf{I}$ and is solvable by a linear system with a least squares algorithm [89]. Therefore, ideally all points should be on the surface of a sphere with radius one, but due to the mismatch of the sensitivities the sphere deforms to an ellipsoid and the bias causes a translation away from the origin. The disadvantage is that without the orientation information the cross-axes alignment cannot be found.

To calibrate the ACC sensor of a hearing aid, we first apply the latter ellipsoid calibration for the sensitivity matching and bias removal. Then, the device orientation is normalized to microphone plane, which is parallel to horizontal world plane. This defines a static orientation normalization to the housing of the HA. To do so, we place the hearing aid in a jig and perform six (up, down, left, right, forward, and backward) reference measurements of known orientation as shown in Fig. 2.9 to measure gravity vector $\mathbf{g}_{\text{up}}$ in the up direction of the world coordinate system [88]. This allows to compute a rotation matrix, which aligns the sensor coordinate system to housing of the HA:

$$\mathbf{R} = \left(\frac{\mathbf{g}_{\text{forward}} - \mathbf{g}_{\text{backward}}}{2} \frac{\mathbf{g}_{\text{left}} - \mathbf{g}_{\text{right}}}{2} \frac{\mathbf{g}_{\text{up}} - \mathbf{g}_{\text{down}}}{2} \right). \tag{2.5}$$

The used static calibration is applied per device and not on a subject basis to address the small individual differences of the wearing orientation, since people always perform small movements and are mostly in the upright position during the day. Therefore, an online orientation calibration is not feasible due to difficult conditions, because the reference orientation is unknown, and the recordings would contain small dynamic accelerations. We use the calibrated sensor values in Equation (2.3) for the further processing.

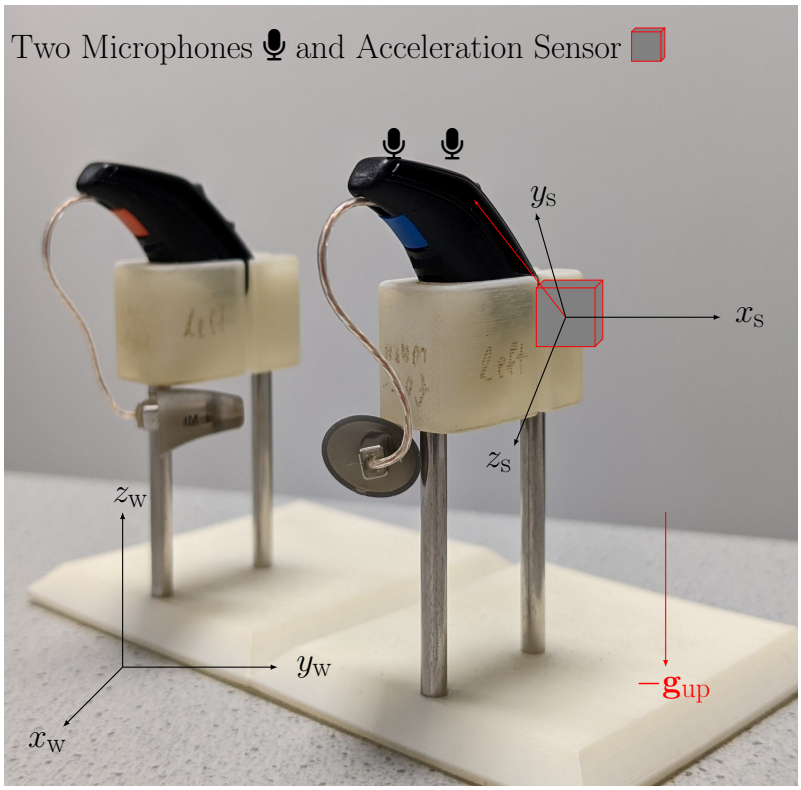


Figure 2.9: The hearing instruments with the two microphones and the built-in accelerometer are mounted in the jig to measure the gravity vector $\mathbf{g}_{\text{up}}$ in the up direction. But the world gravity vector $\mathbf{g}_{\text{w}} = -\mathbf{g}_{\text{up}}$ points towards the earth center, but the static acceleration components are negatively measured. The world and sensor coordinate systems are shown.

2.4.2 Audio Features

For the **audio** signals, hearing aids compute several features to perform the acoustic classification as described in section 1.2. Therefore, a set of 10 precomputed HA features is selected, since they describe various environmental, music, and speech characteristics well, which are helpful to detect the routine classes. The HA transforms the time signal to the frequency domain to compute these features. Thereby, the first band is from 0 to 125 Hz and the remaining 47 channels *ch* have a width of 250 Hz up to 12 kHz. The listed features are grouped by the main detection property.

- The own voice activation takes advantage of the acoustic path from the mouth to the HA microphones [90]. It is very helpful to distinguish social and listening situations. We choose the soft decision, since it is more informative and builds the basis for hard decision by thresholding. It has a scalar value from 0 to 1.

- The absolute value of the correlation coefficient in the following equation between the actual level vector $\mathbf{x}_{n_1}$ of the first 16 channels and the same channel vector a few milliseconds ago $\mathbf{x}_{n_2}$ describes the tonality of music that can differentiate social or listening from other classes [47]:

$$\rho_{\mathbf{x}_{n_1}\mathbf{x}_{n_2}} = \frac{|\sum_{i=1}^{16}(x_{n_1} - \mu_{\mathbf{x}_{n_1}})(x_{n_2} - \mu_{\mathbf{x}_{n_2}})|}{\sigma_{\mathbf{x}_{n_1}}\sigma_{\mathbf{x}_{n_2}}}. \tag{2.6}$$

- Whereas, the wind activity helps to detect outdoor situations, but fast head rotations or movements can also trigger this feature due to the resulting airflow. It uses the non-existing cross-correlation between the front and rear microphone wind signals $\mathbf{x}_{.F}$ and $\mathbf{x}_{.R}$. The scalar feature is computed based on both microphone levels of the first 16 channels:

$$\rho_{\mathbf{x}_{.F}\mathbf{x}_{.R}} = \frac{|\sum_{i=1}^{16}(x_{iF} - \mu_{\mathbf{x}_{.F}})(x_{iR} - \mu_{\mathbf{x}_{.R}})|}{\sigma_{\mathbf{x}_{.F}}\sigma_{\mathbf{x}_{.R}}}. \tag{2.7}$$

- The maximum level

$$\mathrm{lvl}_{\max} = \max_{ch} \mathrm{lvl}_{ch} \tag{2.8}$$

  of all bands $ch \in 1, 2, \ldots, 48$ gives clues about the loudness lvl of the environment and demonstrated a good performance for various audio classification tasks [91].

- The spectral centroid (SC) of noise floor (NF), NF of low- and mid-frequency bands,

$$SC_{NF} = \frac{\sum_{ch=1}^{8} \mathrm{lvl}_{ch} \cdot ch}{\sum_{ch=1}^{8} \mathrm{lvl}_{ch}}, \tag{2.9}$$

$$NF_{\mathrm{Low}} = (\mathrm{lvl}_1 + \mathrm{lvl}_2)/2, \quad \text{and} \tag{2.10}$$

$$NF_{\mathrm{Mid}} = \sum_{ch=1}^{6} \frac{\mathrm{lvl}_{ch} \cdot w_{ch}}{12} \text{ with } \mathbf{w} = [1\,2\,3\,3\,2\,1]^{\mathsf{T}}, \tag{2.11}$$

are good detectors for motorized modes of transportation. The spectral centroid is the frequency-weighted level, which normalized by the overall level. It returns the frequency position as the channel number between 1 and 8. These lower frequencies are relevant to measure the engine or rolling tire noise. That is why, the level of NF in low- and mid-frequency bands is also computed to differentiate low-frequency noise situations from transportation events.

- Three characteristic speech features are the stationarity as the normalized average difference between level and NF, 4 Hertz modulation, and onsets [48, 82].

To summarize all the audio features, we have computed a 10-dimensional feature vector. The three speech features plus the wind, own voice, and correlation features are in interval [0,1]. The three level features NF_{Low}, NF_{Mid}, and lvl_{max} are given in the dB ld domain. The spectral centroid is in the interval [1,8] and represents the channel number, which corresponds a certain frequency position.

2.4.3 Feature Notation

In this section, the feature notation is explained. Scalars are denoted by non-bold letters, whereas multi-dimensional quantities are in bold letters: vectors are in lower-case letters and matrices in upper-case letters. A M-dimensional feature point is expressed in discrete time n with index $n = 1, ..., N$ by the row vector $\mathbf{x}_n = (x_{n1}, x_{n2}, \ldots, x_{nM})$. For all N data samples, it is written as a data matrix $\mathbf{X}$:

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1M} \\ x_{21} & x_{22} & & x_{2M} \\ \vdots & & \ddots & \vdots \\ x_{N1} & \ldots & & x_{NM} \end{pmatrix}, \tag{2.12}$$

which can be also denoted by $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_N)^{\mathrm{T}}$. A corresponding label vector $\mathbf{l} = (l_1, l_2, \ldots, l_N)^{\mathrm{T}}$ is used during the clustering or learning process, where a label l is chosen out of the K classes or clusters $\mathbb{C} = \{c_1, \ldots, c_K\}$ for all of the N samples.

2.5 Data Summary

In this chapter, we introduced the characteristics of the real-world data sets and labels, on which we perform the routine analysis in the following chapters.

Summary

- The discussed common hearing situations and environments allow to derive the intention-based hearing routine annotations for the set $\mathbb{D}_7$.
- The Thomas set $\mathbb{D}_\mathrm{T}$ includes acceleration and audio of a hearing aid with a time diary and offline annotations. It used for the feasibility experiments.
- The real-world set $\mathbb{D}_7$ consists of seven subjects with the same ACC and audio feature like $\mathbb{D}_\mathrm{T}$ set plus intention-based user annotations and allows to analyze the model generalization abilities across subjects.
- The public Huynh set $\mathbb{D}_\mathrm{H}$ consists of two acceleration sensors and serves as an independent evaluation and comparison set.
- The recorded acceleration and audio features of two real-life data sets, $\mathbb{D}_\mathrm{T}$ and $\mathbb{D}_7$, plus the notation are introduced.

Chapter 3

Grouping and Visualizing the Daily Routine Data by Unsupervised Learning

The goal of this chapter is to analyze if the routine data contain groups of similar situations and environments, the so-called clusters, which are important and relevant for the daily routine analysis in hearing aids. To achieve this, we extract features in section 3.1 and apply unsupervised clustering techniques in section 3.2, which do not consider any label information. Afterwards, for the interpretation of these clusters in sections 3.3 and 3.4, we visualize the data embeddings of dimensionality reduction (DR) techniques or analyze the temporal behavior of the found clusters by time schedules. The feasibility analysis is performed on our $\mathbb{D}_{\mathrm{T}}$ set in section 3.5 and the Huynh data is used as a reference for visualization evaluation. The material introduced in this chapter is partially taken from our publication [9].

3.1 Feature Extraction for Clustering

For the clustering analysis on our set $\mathbb{D}_{\mathrm{T}}$, the routine features are designed to represent the recurring routine situations and environments well. We use the input features described in section 2.4 and an overview of the total feature processing is displayed in Fig. 3.1. The space can be partitioned in two independent inputs: ACC and audio. To fuse the sensors on the same time grid, a linear resampling to the stable transmission rate of 8 Hz is executed. The calibrated 3D ACC data and the 10D audio signals build the total 13 inputs for the statistical representation. They are segmented (Segment) in non-overlapping one-minute frames to balance between fast audio (seconds) and slow activity (minutes) changes [41, 92]. Afterwards, the statistical quantities - mean, variance (var), and mean crossing rate (MCR) - are computed for all features and frames [44]. In the context of ACC data, the mean provides the head and body orientation, which is the key identifier to differentiate some scenes [42]. For example, in our case, sitting during office work and laying down in a workout can be distinguished. If we have N_1 samples of one feature $x_1, x_2, \ldots, x_{N_1}$ for one segmentation window, the mean μ is given by

$$\mu = \frac{1}{N_1} \sum_{i=1}^{N_1} x_i. \tag{3.1}$$

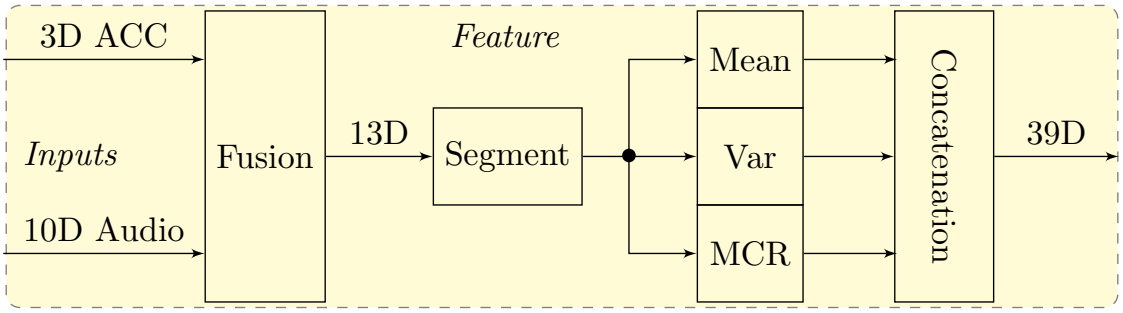


Figure 3.1: The routine feature processing scheme on the set $\mathbb{D}_{\mathrm{T}}$ is depicted, where the audio and acceleration (ACC) inputs at an 8 Hz rate are preprocessed and fused. Afterwards, they are segmented in one-minute frames and statistically summarized (Mean, Variance (Var), and Mean Crossing Rate (MCR)), which gives after concatenation a 39-dimensional data stream.

The variance is a standard measure for motion strength to differentiate actions, such as being at rest, walking, and jogging. It is calculated in an unbiased way by

$$\sigma^2 = \frac{1}{N_1 - 1} \sum_{i=1}^{N_1} (x_i - \mu)^2 \,. \tag{3.2}$$

The authors of [93] demonstrated, that these two simple statistical measures, mean and variance, have a very good performance even against tailored sport features. The MCR informs about the frequency of signal crossing the mean value. It is calculated with the help of the sign function, which is one for non-negative numbers, and minus one for negative numbers:

$$\mathrm{MCR} = \frac{1}{2(N_1 - 1)} \sum_{i=2}^{N_1} \left| \mathrm{sign}(x_i - \mu) - \mathrm{sign}(x_{i-1} - \mu) \right| . \tag{3.3}$$

In the case of HA audio data, the statistical measures deliver a good representation. To sum up, out of 13 inputs three measures are extracted, which gives 39 features for the further analysis.

For the Huynh data set $\mathbb{D}_{\mathrm{H}}$, the recordings are performed by two 3D acceleration sensors, which are worn on the wrist and in the pocket. Due to storage reasons, the measures, mean and standard deviation per feature, are precalculated at a rate of 2.5 Hz, which results in a 12D low-level feature vector. Afterwards, we apply the same three statistical quantities in one-minute frames, which gives a 36D space. The time of day is an optional feature, which gives 37D dimensions and we evaluate its effect on the detection task.

3.2 Clustering the Daily Routine Data

To cluster the daily routine data, we consider the existing clustering techniques, which find natural groupings among the samples or summarize the characteristics of the found groups by representative examples, e.g., for vector quantization. With these methods, we can find out if the routine data contains relevant groups of similar situations and environments. Afterwards, we apply the relevant techniques for the clustering analysis and evaluate their results.

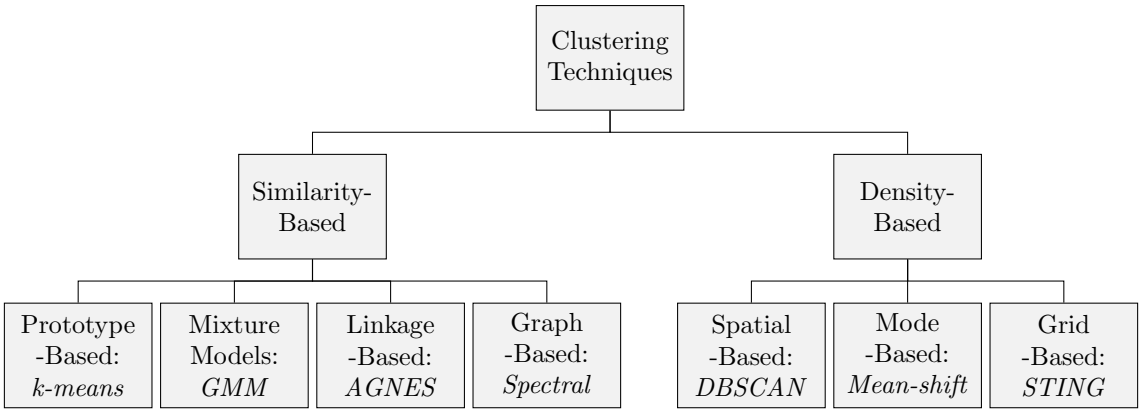


Figure 3.2: The overview of clustering techniques are grouped by similarity-based and density-based methods, where for each subcategory, e.g., mixture models, a prominent example (*GMM*) in cursive is given.

3.2.1 Clustering Techniques

The clustering techniques split up in similarity- and density-based methods. They group the N samples $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ by a measure, which expresses the similarity or dissimilarity of feature points. Alternatively, clusters are assumed to form in area with many points, which is formalized as the density in space. A taxonomy of clustering techniques with different measures is shown in Fig. 3.2 [29]. Due to high number of variants, we briefly explain the most prominent method of each technique group and summarize the key concept.

Similarity Based Methods

The similarity-based methods follow the main principle to minimize the intra-cluster distances while maximizing the inter-cluster distances. The most popular prototype-based algorithm is the **k-means** method [94], where K clusters are represented by the mean of all points within these clusters. The similarity is usually measured by the squared Euclidean distance. The goal is to minimize the overall quadratic distance to all cluster centers $\boldsymbol{\mu}_{c_k}$ for $k \in 1, \ldots, K$ of all clusters $\mathbb{C} = \{c_1, \ldots, c_K\}$, where the within-cluster sum-of-squares criterion is given by

$$\sum_{i=1}^{N} \sum_{\mathbf{x}_i \in c_k} \left\| \mathbf{x}_i - \boldsymbol{\mu}_{c_k} \right\|^2 . \tag{3.4}$$

At the start, the centers are randomly initialized, which can result in poor configurations and clustering outcomes. A better scheme is the k-means++ initialization, which equally distributes the centers across space [95]. Afterwards, the two steps are repeated until the sum-of-squares criterion in Equation (3.4) converges to a minimum. In the first step, each point is assigned to the closest cluster center, and then the centers are repositioned by the mean of all cluster members. The deterministic k-means procedure partitions the cluster space into Voronoi cells. That is why, it is often used for vector quantization or unsupervised feature learning. The biggest disadvantage is the predetermined number of clusters, which needs to be chosen well.

A similar probabilistic technique is the generative **Gaussian mixture model (GMM)**, which defines K Gaussians to model the data as a probability distribution. The parameters are optimized to maximize the a-posteriori probability by the expectation-maximization (EM) algorithm. This is analog to the k-means optimization. Data points are assigned to clusters by a soft probability score instead a hard-assignment like in k-means.

The next approach measures the linkage-based similarity and is a hierarchical clustering method, which is called **agglomerative nesting (AGNES)** [96]. The main advantage is that the number of clusters does not need to be known a priori. At the start, every data point forms one cluster, which makes it a bottom-up approach. Until a stopping criterion is met, the two clusters with the smallest inter-cluster distance are merged. The key point is the similarity (or so-called linkage) measure determining the inter-cluster distance and as a result, which clusters are linked or merged. Multiple options exist, e.g., the minimum (single linkage), average (average linkage), or maximum (complete linkage) distance. The distance metric is often the Euclidean distance. The outcoming hierarchy within the data set is visualized as a dendrogram, which can be cut by a threshold resulting in a cluster configuration.

The graph-based **spectral clustering** constructs a graph measuring the adjacency between all data points, which is represented in the Laplace matrix. Afterwards, it computes the K eigenvectors of the Laplace matrix with the smallest eigenvalues [97]. This embeds the data in a new space, on which a traditional clustering approach like k-means is performed and usually the quality of the outcome is better.

Density Based Models

The density-based clustering methods have the central idea that the cluster borders are areas with a low data density and clusters form areas with a high density. A popular spatial technique is the **density-based spatial clustering of applications with noise algorithm (DBSCAN)** [98]. It has two main parameters, the radius and the minimum cluster size, to define the density as the number of points within the given radius. Thus, a core point in a cluster fulfills the defined requirements, i.e., it has the number of points within the given radius. On the contrary to a border point of a cluster, which has fewer points within the radius, but it is within the neighborhood of a core point. The remaining samples are noise points. The corresponding relationships are illustrated in Fig. 3.3.

The mode-based or mode-seeking algorithm is called **mean-shift** [99]. It searches for the densest area in the neighborhood of a point based a kernel density estimation. Mean-shift moves the cluster center towards this area, which converges to a local density maximum that represents the cluster center. This procedure is done for every data point and all points converge to the local density maximum. Analogically, it is like climbing a mount, where every climber starts around the peak and moves towards the highest point.

The **statistical information grid approach (STING)** creates a grid and counts the density as the number of points within a cell [100]. If this density exceeds a

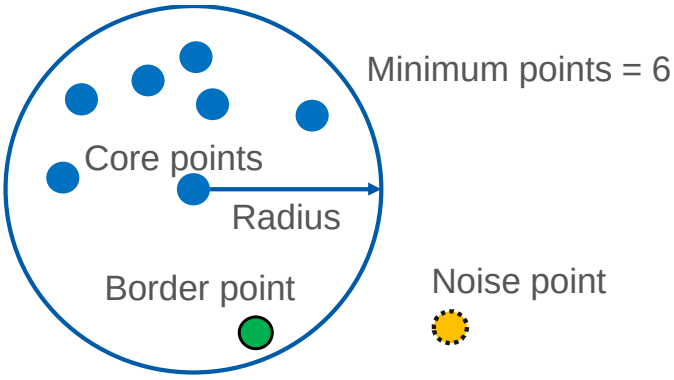


Figure 3.3: The DBSCAN clustering principle shows the relationship of core, border and noise points, where the density is defined as the number of points within a given radius.

threshold, the cell is counted as a cluster. The neighboring cells are checked as well and if they fulfill the requirement, they are merged to the cluster. This expands the cluster until a cell is found, which does not fulfill the density condition.

With these techniques, we can judge if the clustering and manifold assumption is fulfilled, which describe properties of the data distribution [101]. The clustering assumption assumes points in the same cluster have the same label, whereas the manifold assumption states that nearby points on a manifold have similar labels.

3.2.2 Clustering Evaluation Metrics

To assess the quality of a clustering algorithm or the class-separation ability of different embeddings, the clustering validation concepts determine how well the data instances are grouped [102]. They are divided into internal and external measures. The internal measures state the quality of a clustering structure based on the clustering labels and data. The external measures compare the clustering labels to an external ground truth, which does not necessarily exist for all data sets. Therefore, we apply the internal measures to assess the embedding class-separation in section 3.4.2 and the clustering structure of the hierarchical clustering algorithm in section 3.5. For the embedding evaluation, the given Huynh class labels are viewed as output of a clustering algorithm and the following measures are utilized [102].

To compute them, we need to define some helping quantities. Distance measures compare the similarity between two data points, i.e., a low distance value corresponds to a high similarity and vice versa. As defined in [103], the Minkowski distance of two M-dimensional data points $\mathbf{x}_i$ and $\mathbf{x}_j$ is given by:

$$\mathrm{d}_p(\mathbf{x}_i, \mathbf{x}_j) = \sqrt[p]{(x_{i1} - x_{j1})^p + \ldots + (x_{iM} - x_{iM})^p} \tag{3.5}$$

and is known as the $\mathbb{L}_p$-norm. The axes-parallel Manhattan or City-Block distance is $p = 1$, the default Euclidean distance for $p = 2$, and the Maximum distance for $p = \infty$. The measure $\mathrm{d}_p(\mathbf{x}_i, \mathbf{x}_j)$ is greater than zero if $\mathbf{x}_i \neq \mathbf{x}_j$ and zero if $\mathbf{x}_i = \mathbf{x}_j$. The center of the data matrix $\mathbf{X}$ or a cluster c_i is calculated by the mean of all elements: $\boldsymbol{\mu}_{\mathbf{X}}$ or $\boldsymbol{\mu}_{c_i}$. The cluster center is also known as the centroid. The number of

clusters is K and a cluster c_i contains N_i elements. With these helping quantities, we can define the internal clustering metrics to analyze the quality of the grouping:

- **Silhouette** (Silh) analyzes the maximum spread of the overall clustering configuration by computing the pairwise difference of between- and within-cluster distances $b(\mathbf{x})$ and $a(\mathbf{x})$ [104], respectively. Thereby, the between-cluster distance $b(\mathbf{x})$ stands for the separation of groups, whereas the within-cluster distance $a(\mathbf{x})$ represents the tightness of a group:

$$b(\mathbf{x}) = \min_{j,j\neq i} \frac{1}{N_j} \sum_{\mathbf{y}\in c_j} \mathrm{d}_2(\mathbf{x},\mathbf{y}) \text{ and} \tag{3.6}$$

$$a(\mathbf{x}) = \frac{1}{N_i - 1} \sum_{\mathbf{y}\in c_i, \mathbf{y}\neq\mathbf{x}} \mathrm{d}_2(\mathbf{x},\mathbf{y}) \text{ with } \mathbf{x} \in c_i. \tag{3.7}$$

  The silh($\mathbf{x}$) measures the spread per sample. The overall value Silh is the clusterwise-averaged silh($\mathbf{x}$) measure for all data points and accounts for the goodness of the total clustering configuration. Both values range between -1 and +1 and are given by

$$\mathrm{silh}(\mathbf{x}) = \frac{b(\mathbf{x}) - a(\mathbf{x})}{\max\left(b(\mathbf{x}), a(\mathbf{x})\right)} \text{ and} \tag{3.8}$$

$$\mathrm{Silh} = \frac{1}{K} \sum_{i=1}^{K} \sum_{\mathbf{x}\in c_i} \mathrm{silh}(\mathbf{x}). \tag{3.9}$$

  Thus, higher values denote a better grouping and +1 is the best possible outcome.

- **Davies Bouldin** (DB) averages the maximal intercluster-intracluster distance ratios over all clusters [105]. This means for each cluster, the within-cluster distances are computed as the average distance between the cluster centroid and all members. Then, the pair-wise sum of all within-cluster distances is taken. This is set in ratio to the distance between two cluster centroids. Afterwards, the maximum value of the ratio stands for the worst-case configuration for each cluster, which is then averaged over all clusters. Consequently, a lower index means a more compact cluster configuration. The DB value is greater or equal than zero and is computed by

$$\mathrm{DB} = \frac{1}{K} \sum_{i=1}^{K} \max_{j,j\neq i} \left\{ \frac{\frac{1}{N_i}\sum_{\mathbf{x}\in c_i} \mathrm{d}_2(\mathbf{x},\boldsymbol{\mu}_{c_i}) + \frac{1}{N_j}\sum_{\mathbf{x}\in c_j} \mathrm{d}_2(\mathbf{x},\boldsymbol{\mu}_{c_j})}{\mathrm{d}_2(\boldsymbol{\mu}_{c_i},\boldsymbol{\mu}_{c_j})} \right\}. \tag{3.10}$$

- The **SD** (separation (scattering Scat) and inter-cluster distance (Dis)) criterion measures the average spread of points and the aggregated separation of all data groups [106]:

$$\mathrm{SD} = \mathrm{Dis} \cdot \mathrm{Scat} + \mathrm{Dis}, \tag{3.11}$$

  where both terms are given by

$$\mathrm{Scat} = \frac{\sum_{i=1}^{K} \|\mathrm{var}(\mathbf{X}_{c_i})\|}{K\|\mathrm{var}(\mathbf{X})\|} \text{ and} \tag{3.12}$$

$$\mathrm{Dis} = \frac{\max_{j,i} \mathrm{d}_2(\boldsymbol{\mu}_{c_i},\boldsymbol{\mu}_{c_j})}{\min_{j,i} \mathrm{d}_2(\boldsymbol{\mu}_{c_i},\boldsymbol{\mu}_{c_j})} \sum_{i=1}^{K} \frac{1}{\sum_{j=1}^{K} \mathrm{d}_2(\boldsymbol{\mu}_{c_i},\boldsymbol{\mu}_{c_j})}. \tag{3.13}$$

The scattering term expresses the compactness as the ratio between the average cluster compactness divided by the total compactness of the data. The variance is used as a measure for the compactness and is calculated per feature column as shown in Equation (3.2). The $\mathbf{X}_{c_i}$ denotes all points within cluster c_i. The inter-cluster distance is computed as distance between the cluster centroids. Therefore, the lowest SD value represents the best cluster separation, and the value is greater or equal than zero.

- The **distance matrix D** computes the Euclidean distances between the single data points and the main diagonal is zero:

$$\mathbf{D}(\mathbf{X}) = \begin{pmatrix} \mathrm{d}_2(\mathbf{x}_1, \mathbf{x}_1) & \mathrm{d}_2(\mathbf{x}_1, \mathbf{x}_2) & \dots & \mathrm{d}_2(\mathbf{x}_1, \mathbf{x}_N) \\ \mathrm{d}_2(\mathbf{x}_2, \mathbf{x}_1) & \mathrm{d}_2(\mathbf{x}_2, \mathbf{x}_2) & & \mathrm{d}_2(\mathbf{x}_2, \mathbf{x}_N) \\ \vdots & & \ddots & \vdots \\ \mathrm{d}_2(\mathbf{x}_N, \mathbf{x}_1) & \dots & & \mathrm{d}_2(\mathbf{x}_N, \mathbf{x}_N) \end{pmatrix}. \tag{3.14}$$

The cluster-wise ordered matrix shows a block pattern of areas with low (intra-cluster) and high (inter-cluster) distances if the clustering groups the samples well.

3.3 Visualizing the Daily Routine Data Temporally

To gain an understanding of the data set characteristics, visualization techniques are applied [33]. The chosen technique depends on the data attributes, e.g., number of dimensions, temporal or numeric data [34]. A few non-exhaustive examples are line graphs, scatter plots, circular or parallel coordinate displays [35].

We plot the multi-dimensional features as a line graph of an exemplary working day from our data set $\mathbb{D}_\mathrm{T}$ in Fig. 3.4. At the start and end of the recording, at 10 a.m. and 3 p.m., the bicycle was used for commuting, which is expressed in the acceleration patterns, increased noise floor features and the wind activity. A similar behavior occurs during lunch time at 12 a.m., but the acceleration patterns of walking to the canteen differ to the cycling activity and the noise floor reaches its highest value in the loud canteen environment at this day. Additionally, a conversation is held, which activates the own voice feature, and the speech components reduce the stationary properties of the audio signal. During the office work, short conversations occurred, for example, at 10.30 a.m., where head rotations produce the dynamic shifts around the gravitational level, and we notice changes in the acoustic levels.

To show the time behavior of multiple days and make it easy to compare the differences and similarities within these days, the multi-dimensional feature plot in Fig. 3.4 is not able to visualize the data in a perceivable manner. That is why, we plot a time schedule of several groups, e.g., classes or clusters, in Fig. 3.7 (a) and explain the time diary annotations in section 3.5. This allows to assess the temporal regularities of the routine behavior. Further possibilities are showing color maps or histograms over time. We plot the activity loudness histogram of data set $\mathbb{D}_\mathrm{T}$ in Fig. 3.5. The activity strength is calculated as the variance of acceleration vector norm and is quantized into three categories: low, medium, and high (l, m, and h).

This means in the case of sitting the acceleration vector norm is about 1g and has a small variance. In contrast to walking, where the norm strongly changes, i.e., it results in a high variance. The loudness as the max. level feature is quantized into three categories: quiet, medium, and loud (Q, M, and L). To compare the quantities over different days, we show the frequency of the 9 combinations such as lQ meaning a low activity strength in a quite environment. The lQ bin has often a high value during office work. As shown Fig. 3.4, the lunch break situation is highly notable in day 1 due to loud environment. This behavior repeats over multiple days (1, 5, 7, 9). Conversations are mainly differentiated from the office work by the increased loudness, since the activity only slightly enhances. The running exercise at 3 p.m. on Sunday has a high activity strength with a medium loudness, whereas the car ride at 6 p.m. has is very loud with a medium ACC strength. In contrast to watching TV on the evening at day 2 and 6, they have a low to medium activity and a quiet to medium loudness. Thus, several situations are differentiable within the visualized feature space, and we notice the recurring temporal behavior of routine activities.

3.4 Visualizing the Daily Routine Data Spatially by Manifold Learning

After building the feature representation in 3.1, the data has 39 dimensions, which makes it very difficult for the visualization and computation of distance metrics [35, 107]. That is why, it must be reduced to two (or three) dimensions. We give an overview of well-performing dimensionality reduction (DR) methods and evaluate a number of techniques.

3.4.1 Dimensionality Reduction and Manifold Learning Techniques

Various DR techniques exist, and a comparative overview can be found in [36]. These methods apply linear or non-linear transformations to reduce the data dimensions while showing the inherent structure. One central idea is that the data lies on or near a lower dimensional manifold. The **t-distributed stochastic neighbor embedding (t-SNE)** is applied to visualize the high-dimensional data $\mathbf{x}_1, \ldots, \mathbf{x}_N$ as 2D points $\mathbf{y}_1, \ldots, \mathbf{y}_N$ [37]. It minimizes the Kullback-Leibler-divergence between the two probability distributions of high-dimensional Gaussian and low-dimensional Student-t kernel. Based on a gradient descent optimization, the low-dimensional points are moved in space to achieve the highest similarity between the two probability distributions.

The found mapping is non-linear and data-adaptive, i.e., the technique performs distinct transformations on different regions [108]. This is a result from the binary search of σ_i for each point to ensure that the effective number of neighbors is 50 being equal to the perplexity parameter. In the low-dimensional space, through the heavy-tailed Student-t distribution the similarity to the neighbors is established, since for distant points the probability is infinitesimal small. This means that embedded points separated by gaps are not similar even if they are spatially close. In that way, t-SNE accomplishes a good performance in terms of keeping the local and global structure of the data. Moreover, the low-dimensional representation needs an

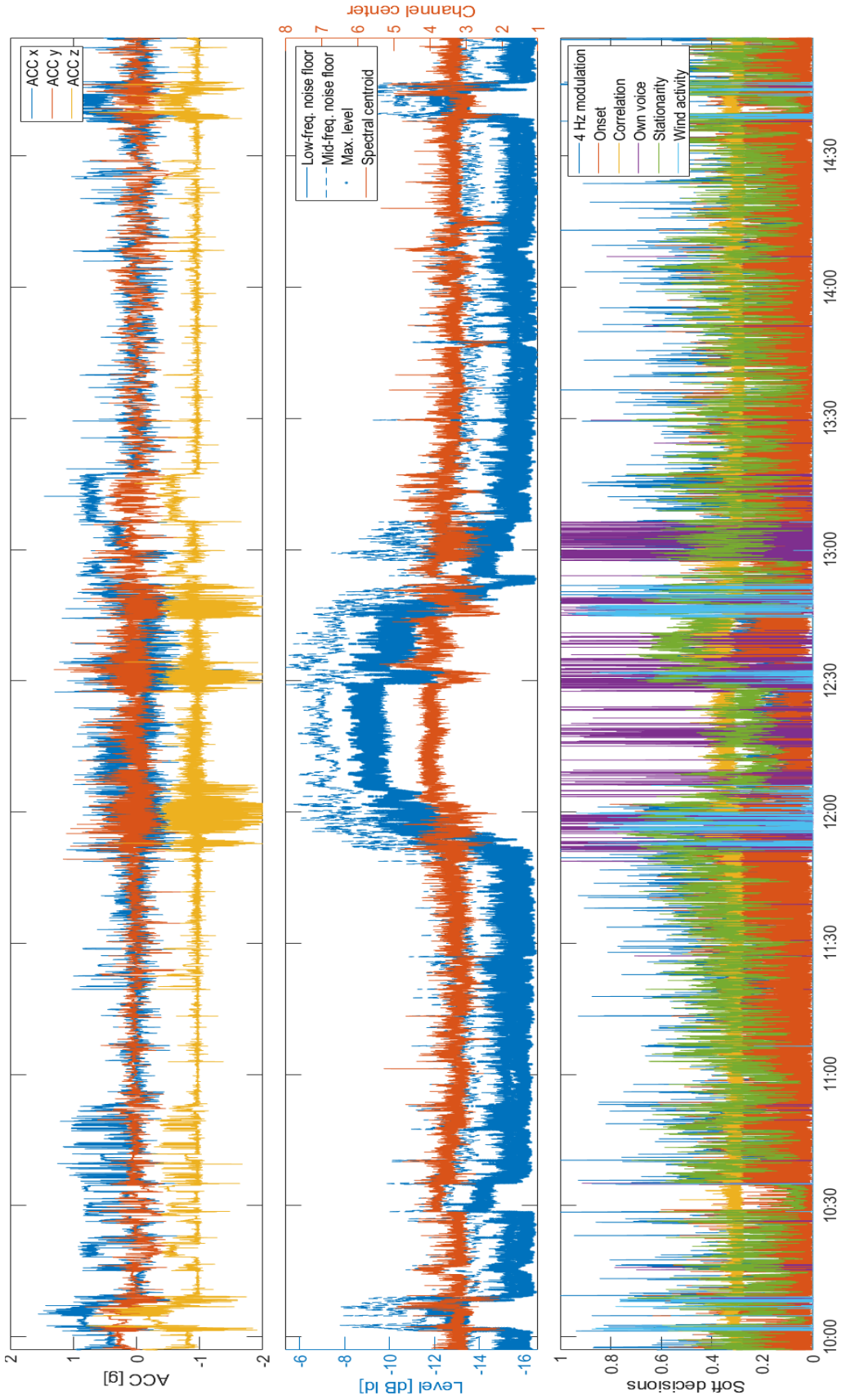


Figure 3.4: The features over an example day (number 1) of our data set $\mathbb{D}_{\mathrm{T}}$.

Figure 3.5: The activity loudness histogram of our data set $\mathbb{D}_{\mathrm{T}}$ represents the normalized duration in each activity loudness bin over one minute. The acceleration activity in the first row is quantized into three categories: low, medium, and high (l, m, and h). The loudness in the second row is quantized into three categories: quiet, medium, and loud (Q, M, and L). The white-colored bins represent a value of zero to increase the readability.

initialization before the optimization starts and it is sampled from a 2D Gaussian distribution with zero-mean and a diagonal unit covariance matrix scaled by 0.0001. To eliminate the effect of bad initialization, multiple runs are performed, and the best embedding is taken as output [109].

Further DR techniques are tested for 2D visualization [36]. The **principal component analysis (PCA)** finds a linear transformation of the high-dimensional data, which maximizes the amount of variance described by the data. This linear mapping is described by the principal eigenvectors, the so-called principal components, with the biggest eigenvalues of the covariance matrix.

Isomap addresses the problem that data might lie on a (e.g., curved) manifold and the Euclidean distance is an inappropriate measure for the pairwise distance between two samples on a manifold [110]. The reason is that the Euclidean distance measures the shortest distance corresponding to the direct connection between two points. Therefore, the geodesic distance is computed expressing the distance on the manifold by a constructing a neighborhood graph and calculating the distance along the graph. Afterwards, the eigenvalue decomposition finds the transformation matrix out of the eigenvectors with the biggest eigenvalues.

The **Sammon mapping** is a non-linear projection technique [111], which preserves the inter-sample distances between the high-dimensional and low-dimensional point pairs, the so-called Sammon error. This should ensure that the same data structure is preserved in both spaces. The low-dimensional points are usually initialized by a PCA transformation, and then the points are iteratively moved in space to mimimize the Sammon error by a gradient descent optimization.

The **locally linear embedding (LLE)** combines several locally linear models to fit the overall global data structure. It can be thought as a number of local PCA transformations, which are combined for a global data projection. To do so, LLE makes a nearest neighbor search for the construction of the weighted local neighborhoods to compute a local eigenvalue decomposition for each local transformation.

The **uniform manifold approximation and projection (UMAP)** algorithm assumes that the data lies on a locally connected manifold [112]. It constructs a neighborhood graph based on the number of nearest neighbors and a minimum distance between two points in the low-dimensional space. The minimum distance stands for the tightness how close the embedded points are put together. UMAP shows to be competitive for visualization purposes in comparison to t-SNE, since it also preserves the global structure well. Depending on the data set and initialization, the performance is similar [109]. The main advantage is, that the embedding is fast and scalable. Both DR techniques warp the space, which means distances are less meaningful compared to PCA. All techniques are part of the toolbox except the UMAP technique, in [36], using the suggested settings and t-SNE performed better by a perplexity of 50 [108].

3.4.2 Evaluating the best Dimensionality Reduction and Manifold Learning Technique

To find the best embedding, we assess the informative value of the representation on our set $\mathbb{D}_{\mathrm{T}}$. That is why, the t-SNE embedded points are plotted with color-encoded high-dimensional features as depicted in Fig. 3.6 for the mean of maximum loudness feature. The spatial grouping of different routines in the t-SNE embedding

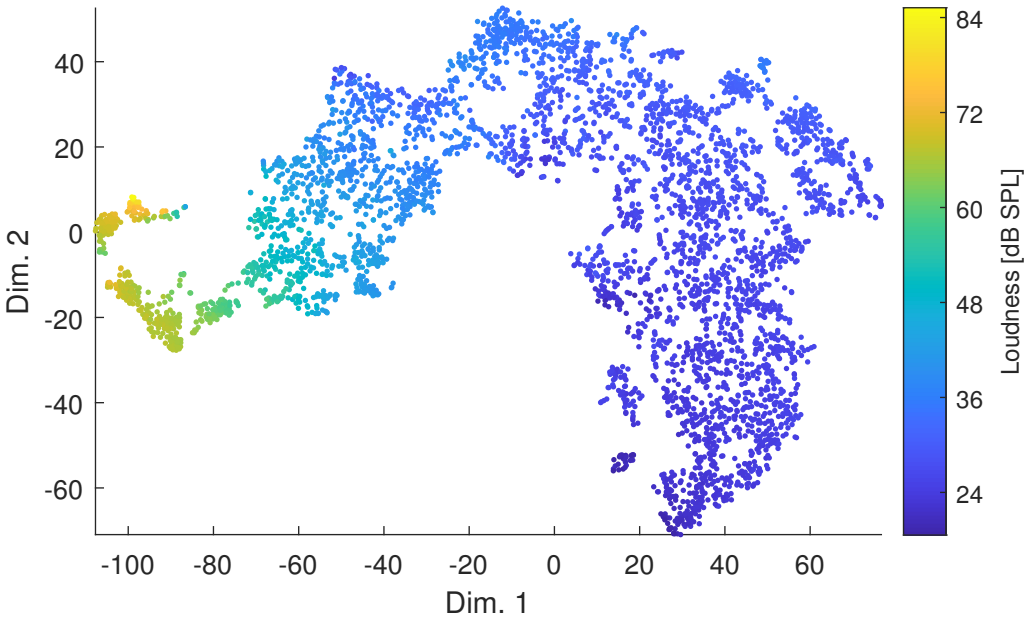


Figure 3.6: The t-SNE embedding of our data set $\mathbb{D}_{\mathrm{T}}$ with color-encoded mean of the maximum loudness feature in the dB SPL domain.

Table 3.1: Embedding evaluation on $\mathbb{D}_{\mathrm{H}}$ set with the Huynh labels as the clustering output. The range plus the optimal value (the smallest (↓) or highest (↑)) of each clustering metric are given.

	t-SNE	**Isomap**	**LLE**	**PCA**	**Sammon**	**UMAP**	**Range**
Silh ↑	-0.03	-0.03	-0.09	**-0.02**	-0.03	**-0.02**	[-1,1]
DB ↓	**4.83**	13.44	15.27	10.06	12.03	7.73	[0,∞(
SD ↓	**10.39**	31.80	66.78	20.71	33.40	22.81	[0,∞(

is analyzed with the hierarchical clustering in section 3.5. Thus, the loudness feature shows to be a characteristic property that has a strong influence on the spatial grouping of the data points. The softest situations and environments are placed at the positive side of the first embedding dimension, whereas loud ones are grouped in the negative direction. Other features, which are not shown as the color-encoding of t-SNE space, also contribute to the spatial grouping of the data points in the embedding. Some attributes only have different values in local areas, which means they differentiate only specific situations, whereas others play a global role in the data grouping. For example, the wind is only active in a small subset of situations. Thus, the corresponding feature has mostly a value of zero during the recordings, but it helps to explain the situation in a few outside environments. In contrast to the wind occurrence, the variance of ACC data has a global influence on the grouping, since it provides valuable information in multiple situations. Further embeddings are plotted in Fig. 4.3 and show a less structured ordering of routine behavior.

To assess the quality of the embeddings with the cluster coefficients in section 3.2.2, the Huynh data serves as an independent validation set. Thus, the given ground truth (GT) is taken as the clustering output and the cluster coefficients of different embeddings are computed in Table 3.1, where the best outcome is marked in bold, and the possible range of metrics is given. For DB and SD, the smallest value is the best, whereas the highest value is the best for Silh. Therefore, t-

SNE outperforms the other representations in two of three metrics, and only at the Silhouette coefficient PCA and UMAP are slightly better, but in this metric the difference is very small between all embedding except for the LLE. In particular, in the DB and SD cluster evaluation coefficients, t-SNE has a strongly smaller value up to many times. This outcome is also visually confirmed as shown in Fig. 4.4.

In short summary, t-SNE is the best embedding and is taken for the visualization.

3.5 Confirming the Clustering and Manifold Assumptions

To analyze the spatial and temporal grouping behavior, we consider the clustering techniques in section 3.2.1 and tested several configurations plus methods, e.g., k-means, DBSCAN, and AGNES. Finally, we choose the AGNES hierarchical clustering analysis, since we do not know a priori the number of clusters and AGNES returns a dendrogram, which visualizes the spatial data structure. The horizontal lines in a dendrogram represent the linkage distance to merge two subgroups into one. Thus, a high linkage distance means the subgroups strongly differ in the feature characteristics. We perform the clustering on the t-SNE space of our set $\mathbb{D}_{\mathrm{T}}$ with a Euclidean distance and average linkage measure. The results for four clusters are displayed in Fig. 3.7, where the cutting level (red line) of the dendrogram in Fig. 3.7 (c) is shown and all connected branches below the threshold form one color-encoded cluster. We show the full dendrogram to make existing subgroups in one cluster visible. A lower cutting level would lead to more clusters with a smaller number of samples in each cluster and vice versa. Too many clusters make it hard to detect the recurring behavior over time. We tested multiple values and chose the threshold, where the clusters represent the temporal structure of the time diary annotations in Fig. 3.7 (a). Thus, the recurring situations and environments fall into the same clusters over different days.

Obviously, comparing the loudness-encoded t-SNE embedding in Fig. 3.6 with the marked clusters in Fig. 3.7 (b), we notice a strong link between the loudness feature and the clustering outcome since the clusters are strongly differentiated by the loudness feature. Other features also contribute to the spatial grouping and clustering by differentiating the different situations and environments. In the following, we explain the cluster structure based on the distinguishing example of the loudness characteristics. The green cluster 2 corresponds to relatively quiet situations mainly like office work in a quiet room. The red cluster 4 contains louder scenes, such as conversations during meetings, working in a busy environment, or watching TV. The blue cluster 3 includes the walking to the canteen at day 1, the running exercise at day 2, or the general louder environments than in clusters 2 and 3. The lunch event in the canteen environment has a very loud babble noise and is consistently detected by the same purple cluster 1. Further included noisy scenes are the car ride at day 2 or the power workout at day 7 with motivating bass music. The car ride, in particular, forms an own subcluster within the purple group, which is also visible in the dendrogram in Fig. 3.7 (c). Of course, the spatial grouping is also determined by all other features than the loudness as described in section 3.4.2. For example, the conversations in cluster 4 are distinguishable from the office work of cluster 2 by the own voice activation or other speech features.

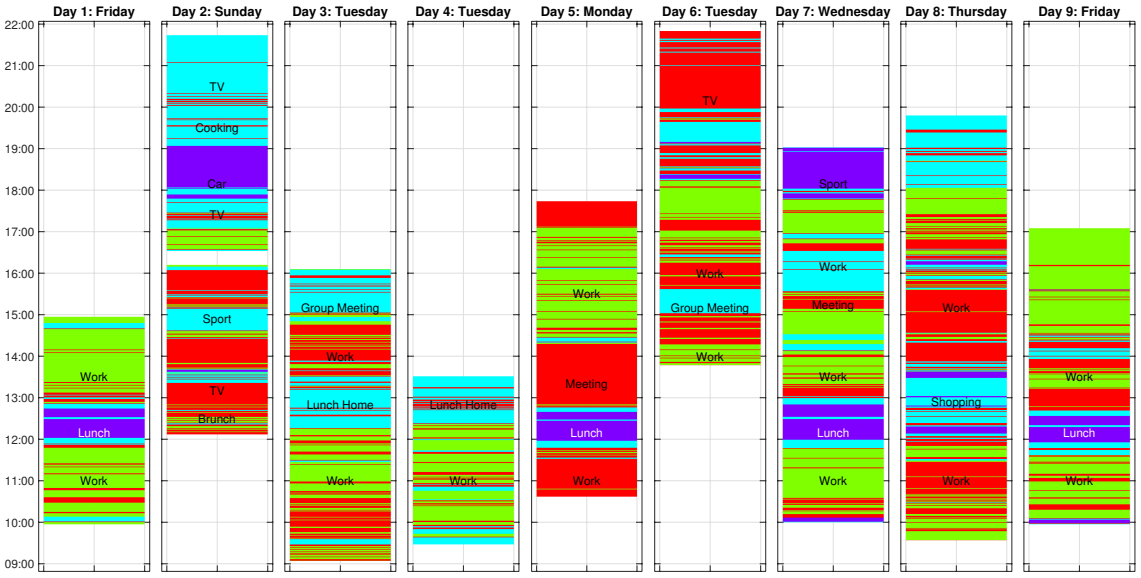
To evaluate the clustering quality as explained in section 3.2.2, we use the ordered distance matrix and silhouette coefficient per sample and cluster in Fig. 3.7 (d) and 3.7 (e). The **distance matrix** shows the dissimilarity between the feature points and silhouette coefficient explains how well a sample fits in its own cluster. Both measures are computed on the high-dimensional features, since we want to show that the t-SNE distances are relevant and correspond to the high-dimensional data structure. In Fig. 3.7 (d), the distance matrix contains four distinct blocks of low pair-wise distances and between them areas of high distances occur. This block pattern stands for well grouped clusters. For example, the cluster 1 has the highest dissimilarity with clusters 2 and 4, which are similar to each other. Thus, the clusters form relevant and distinct groups, since the distance matrix has a clear block structure. The sample-wise **silhouette coefficients** state a strong similarity within clusters 1 and 2, because nearly all members of both clusters have a high silhouette value over 0.7. In contrast to clusters 3 and 4, where a part of the samples has a more distinct feature behavior, since two subgroups exist within the two clusters as shown in Fig. 3.7 (c). This is demonstrated by the negative silhouette coefficients for the subgroups within clusters 3 and 4, which is also expressed in the high linkage distance of the dendrogram. The clustering evaluation states that the clusters are representative for the high-dimensional data structure, but in two clusters valid subgroups exist.

In Fig. 3.7 (a), we further notice the recurring patterns of the daily routine, such as similar lunch break times, the group meeting on Tuesdays at 3 p.m., commuting events, or working periods. These activities have a strong temporal structure and have a longer duration up to hours. In contrast, e.g., spontaneous conversations in red cluster 4 during the working periods do not follow a certain order and can happen at any time but have a short duration of minutes. Thus, we found relevant clusters for the temporal structure of the daily routine.

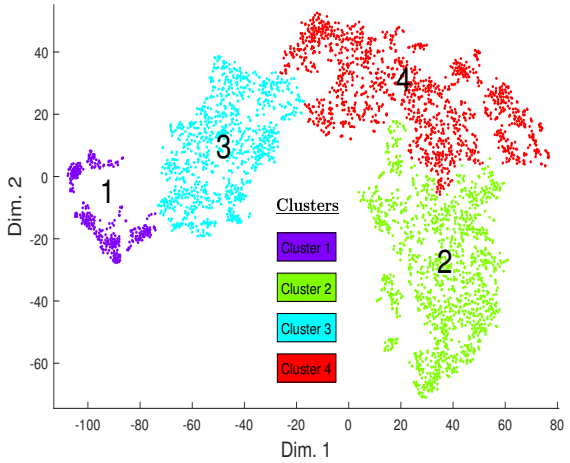
From these observations, we state a good detection of some characteristic situations, such as the lunch in the canteen, car, TV, or workout. These scenes are very distinct and form own clusters within our feature space. This confirms the clustering assumption that the data set contains distinct situations, which are recognizable by the features. The manifold assumption is fulfilled for the connected clusters 2, 3, and 4, since nearby routine situations have similar characteristics such as the working in a calm environment.

3.6 Grouping and Visualization Summary

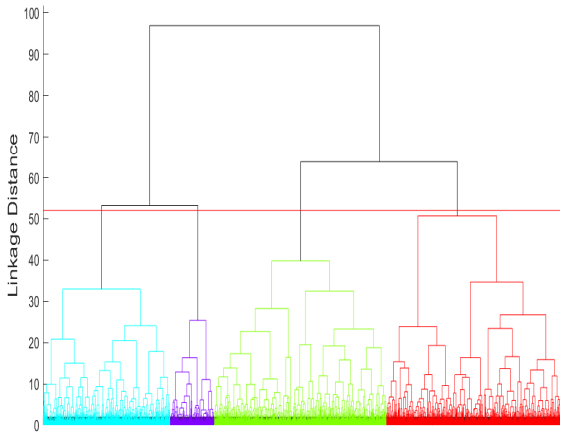
In this chapter, we used unsupervised methods to visualize and group the daily routine situations and environments of our data set $\mathbb{D}_{\mathrm{T}}$. To do so, we built a statistical feature representation and performed a clustering method on a low-dimensional data embedding, which we validated with clustering evaluation techniques. To validate the dimensionality reduction methods for the visualization purposes, we applied our approach on the public data set $\mathbb{D}_{\mathrm{H}}$.



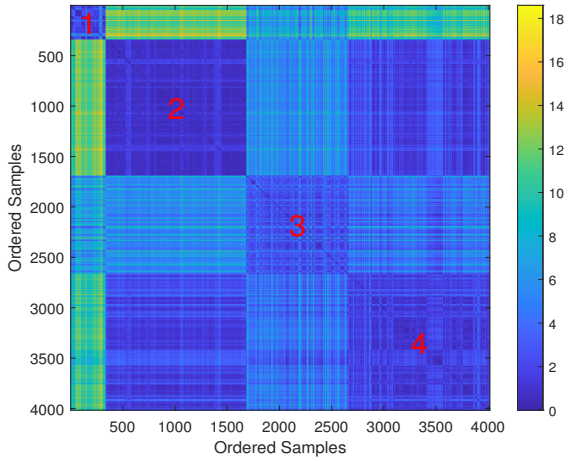
(a) Clusters over time with the time diary annotations.



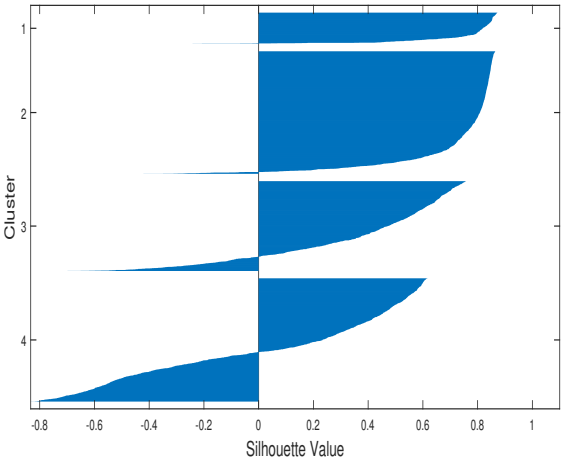
(b) 2D t-SNE embedding.



(c) Dendrogram with the cutting level (red line).



(d) Distance matrix sorted per cluster.



(e) Silhouette coefficient silh sorted per cluster.

Figure 3.7: The results of hierarchical clustering on the data set $\mathbb{D}_{\mathrm{T}}$ are shown as color-encoded clusters over time in (a), in the t-SNE embedding in (b), as dendrogram in (c), cluster-wise ordered samples in a distance matrix in (d), and sample-wise silhouette coefficients sorted per cluster in (e).

Summary

- We evaluated various dimensionality reduction techniques and the t-distributed stochastic neighbor embedding (t-SNE) found a very meaningful embedding of the high-dimensional data.
- We performed the hierarchical clustering approach on the 2D embedding. With the help of the visualization techniques, we have shown that some characteristic situations, such as the lunch in the canteen, car, TV, or workout are distinguishable.
- These distinct scenes form own clusters within our feature space, i.e., we confirm the clustering assumption that our data contains distinct situations, which are recognizable by the features.
- The manifold assumption is fulfilled, since nearby routine situations have similar characteristics and are locally grouped together.

Chapter 4

Labeling the Daily Routine Data by Semi-Supervised Learning

In this chapter, the daily routine of the data set $\mathbb{D}_{\mathrm{T}}$ with the time diary annotations is labeled and recognized by semi-supervised learning techniques. Therefore, we use during the labeling process the gained knowledge of chapter 3 that the situations and environments cluster in a relevant temporal structure on our feature representation explained in section 3.1. The semi-supervised processing scheme is introduced in Fig. 4.1, which includes the two steps: propagating the labels in section 4.1 and the daily routine recognition (DRR) in section 4.2. An overview of semi-supervised label technique is given in section 4.1.1, the extended visual interactive labeling (VIL) is introduced in section 4.1.2 and is applied on our set $\mathbb{D}_{\mathrm{T}}$ in section 4.1.3. Afterwards, based on these annotations the classifiers are trained in section 4.2.1, the evaluation metrics are introduced in section 4.2.2, and the performance is assessed in section 4.2.3. Furthermore, we compare the routine classification performance of different input data variants to evaluate the impact of the sensor modalities, ACC and audio. The Huynh data set $\mathbb{D}_{\mathrm{H}}$ serves as a reference for the evaluation of the VIL procedure. The material introduced in this chapter is partially taken from our publication [9].

4.1 Solving the Labeling Problem

To solve the labeling problem of our routine data $\mathbb{D}_{\mathrm{T}}$, semi-supervised learning techniques are applied. We give an overview of existing approaches and extend the VIL method to annotate the data set. The Huynh set is used as a reference for evaluation purposes.

4.1.1 Labeling Techniques

In semi-supervised learning, a model is trained on a small subset of labeled data and this knowledge is transferred to the remaining larger set of unlabeled points. VIL combines a state-of-the-art dimensionality reduction technique with the human perception [38]. In a nutshell, the algorithm works as follows: the user selects objects in low-dimensional representation and labels them. Afterwards, the classifier is trained on the manually chosen data points and predicts the remaining objects. This procedure is repeated until the result is visually satisfying.

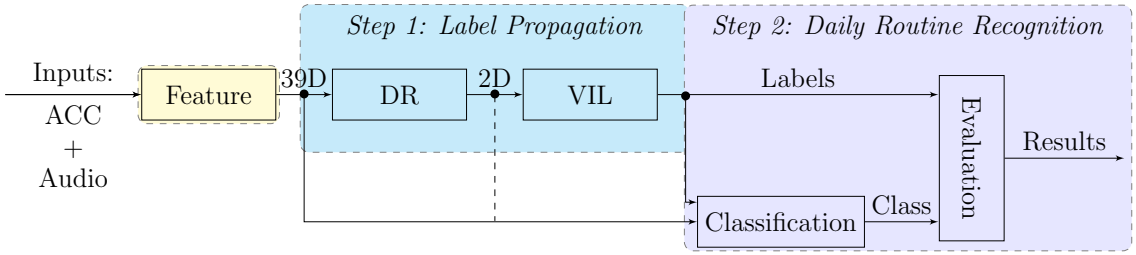


Figure 4.1: The semi-supervised processing scheme follows two steps: propagating the labels and recognizing the daily routine. To do so, the feature representation is built on the audio and acceleration (ACC) inputs, giving a 39-dimensional data stream. To visualize and annotate the data, the DR (dimensionality reduction) stage extracts two dimensions. Then as a first step within the visual interactive labeling (VIL) scheme, the user interactively labels the 2D data and a first classifier is trained to propagate these labels from certainly known events to all other samples. Using these annotations, a second classifier is trained within a leave-one-fold-out (LOFO) cross-validation scheme, and the following evaluation determines the results.

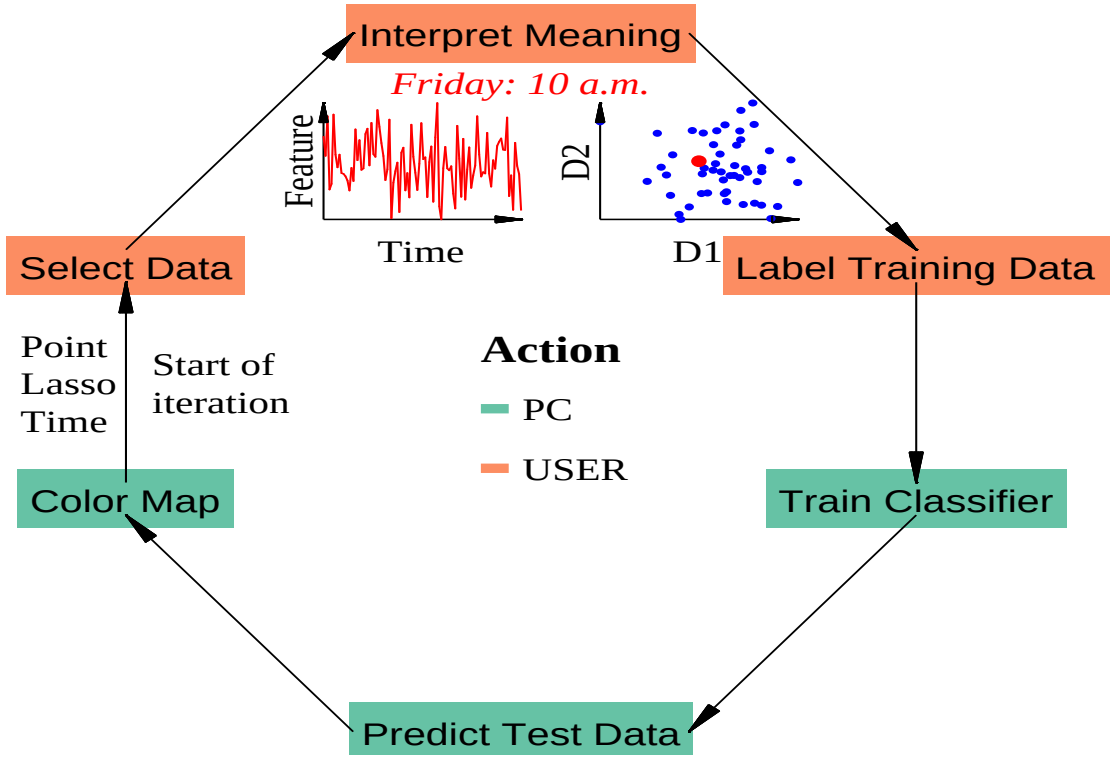
In contrast to VIL, in active learning selects an algorithm the objects, which best optimize the learning model and asks the user to label them [38]. Different strategies exist to select the objects, contributing the most to an improved learning model [39]. One strategy is called *uncertainty sampling*, i.e., an object is selected, where the model is the most uncertain about. Thereby, distance-based classifiers like k-nearest neighbor choose objects, which lie close to the decision boundary. Another strategy is the *error reduction*, where objects are selected that reduce the training or generalization error the most. A third strategy is called *query-by-committee*, meaning an ensemble classifier judges the labeling quality and the instances are chosen with the highest disagreement among the ensemble.

Both semi-supervised methods work well for annotating high-level activities and daily routines with time diaries. Without too much labeling effort, high-quality annotations are generated by a coarse collection of known activities and routines [40].

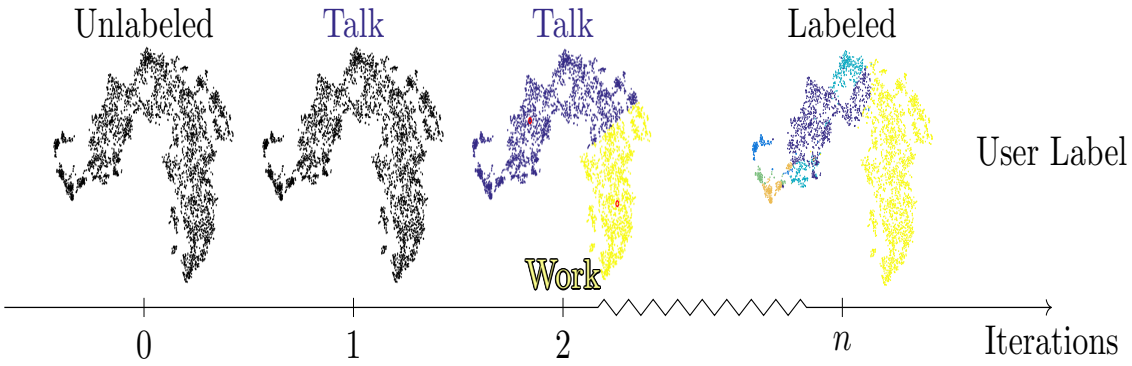
4.1.2 Visual Interactive Labeling Technique

VIL annotates our data set $\mathbb{D}_{\mathrm{T}}$ and the reference set $\mathbb{D}_{\mathrm{H}}$ by propagating the labels from certainly known time diary events to all remaining ones [38]. It operates on a two-dimensional t-SNE embedding of our feature representation explained in section 3.1. The used dimensionality reduction (DR) techniques are introduced in section 3.4. Additionally, further improvements of VIL for the activity recognition challenges are demonstrated. The VIL workflow is illustrated in Fig. 4.2 with the scheme of one iteration in (a) and the annotations over several iterations in (b). The scheme is divided in PC and user actions. At first, the unlabeled t-SNE data is plotted with the same color for all points (iteration 0). Then, the semi-supervised labeling scheme begins and iteratively refines the annotations and the corresponding coloring of the 2D points within a graphical user interface (iterations: $1, 2, \ldots, n$).

At the start of each iteration, as a user we select single or multiple instance/s with a lasso tool, a time or point selector. Hereby, the human visual selection strategies



(a) Extended VIL scheme.



(b) VIL annotations over iterations.

Figure 4.2: The extended visual interactive labeling (VIL) workflow is depicted in (a) for each iteration, where the user selects and interprets the data to label certainly known instances. Afterwards, the automatic PC actions start to train a classifier on the labeled samples and predict the unlabeled instances. Then, the color encoding of the classes is adapted to the predictions as shown over the iterations in (b). At the start, the unlabeled 2D data are drawn in black, the user selected training samples are marked with two red circles (o) and the predicted samples are color-encoded.

are intuitively applied to the two-dimensional scatter plot [38]. These techniques label the map in equal spread, dense areas-, centroids- or outliers-first.

Before we can label the training data, we **interpret the meaning** of chosen points corresponding to the routine behavior. Since every sample in the 2D space represents a high-dimensional curve, the 13 inputs are separately plotted over time as exemplary shown in Fig. 4.2. This is a big advantage against only considering the reduced dimensions, which loses the interpretability through the DR. Therefore, we can infer the temporal characteristics and tie them to spatial position in the embedding space. Thus, the extended VIL framework is perfectly capable for **data exploration** purposes.

Further improvements are achieved by including the time information. This means the current labels are also illustrated in time schedule for each day as shown in Fig. 4.3 (c). Additionally, the selected points are marked, or their timestamps are displayed. Thus, it allows to **assess the temporal behavior**, i.e., if all points occur in irregular time instances, within one or multiple intervals. This gives the big picture over the routine occurrences. Moreover, in combination with a time diary, we select instances based on a time interval, which certainly consist of one activity. Afterwards, these points are marked in the 2D map, and the consistency of selection is easily demonstrated by the proximity of chosen samples.

After two label categories are selected as red dots shown in iteration 2 in Fig. 4.2 (b), a **classifier**, the Gaussian kernel SVM with the same configuration as in 4.2.1, is trained on the chosen and labeled samples. Subsequently, it is applied on remaining (test) samples and the coloring is adapted to the classifier predictions. Then, these steps are redone until the results are visually satisfying and the time schedule matches the coarse time dairy. Thus, the final annotations are obtained for the classification task in section 4.2.

4.1.3 Visual Interactive Labeling Results

Starting with the labeling process on our data set $\mathbb{D}_{\mathrm{T}}$, the analysis of the t-SNE and VIL outcomes is performed in Fig. 4.3 (a), where the VIL annotations are color-encoded. Other views of DR techniques are shown in Fig. 4.3 (b) to gain more insights, but they do not turn out to be as meaningful as t-SNE. They focus on retaining large pairwise distance by relying on the data covariance as PCA or LLE. Likewise, large geodesic distances of Isomap are short-circuited by noise [36].

After the VIL annotation process, a clear structure within the t-SNE map is visible in Fig. 4.3 (a), i.e., a continuous transition is obvious between similar routines such as work and talk, which underlines the usefulness of embedding in terms of keeping similarities. For example, some points of transport routine are embedded within the talk routine, i.e., the points are grouped by the dominating conversational characteristics due to the higher similarity to these neighboring points. This shows a drawback of the labeling or classification with a hard assignment.

Moreover, the map encodes the loudness of the situations. These range from quiet ones (e.g., work), to situations with medium loudness (e.g., talk), to louder ones (e.g., sport with gym music). To make this obvious, the loudness is plotted as color information of the t-SNE space in Fig 3.6. That is why, the two smaller clusters on the left correspond to the transport routine with a stronger low-frequent noise. To assess the consistency between the VIL annotations and time diary, the labels

Table 4.1: VIL and DRR results on Huynh set plus a comparison to previous Huynh DRR results [%].

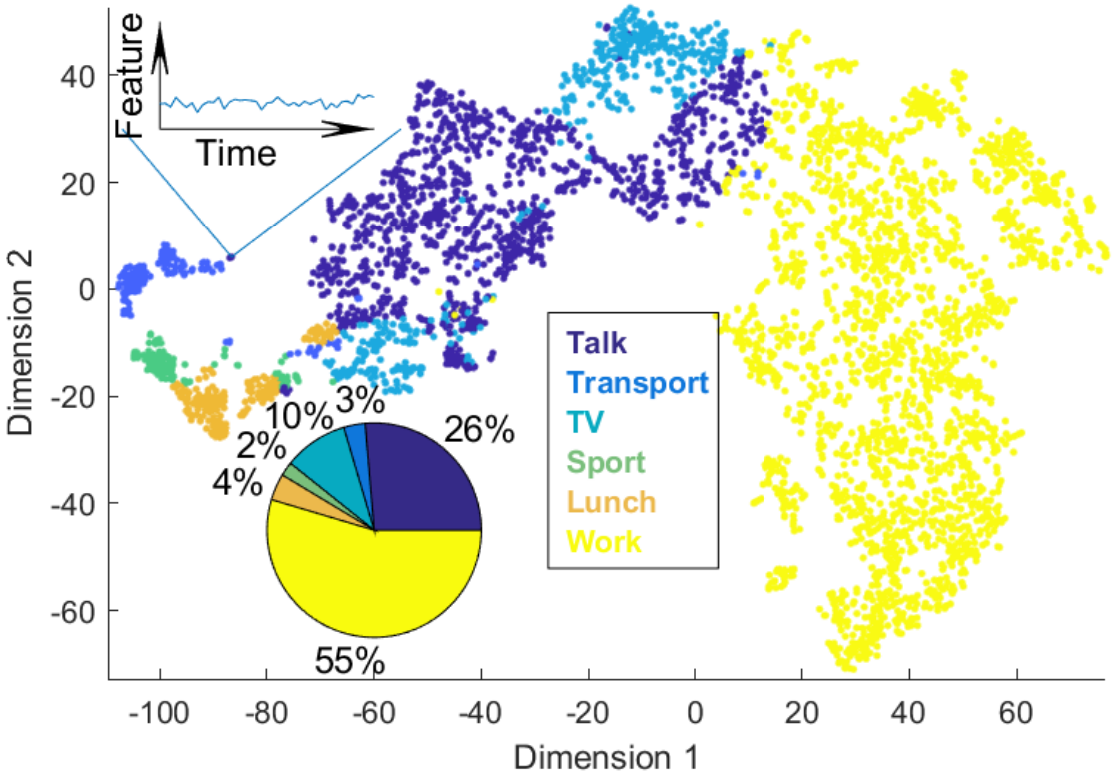
	Dinner	**Commuting**	**Lunch**	**Work**	**Mean**	**Huynh**	**Task**
Precision	15.3	70.5	31.1	92.0	52.2		
Recall	22.9	50.2	64.6	87.1	56.2		**VIL**
Accuracy					79.5		
Precision	82.0	88.4	89.9	94.7	88.7	86.1	
Recall	84.3	85.1	70.2	97.7	84.3	67.2	**DRR**
Accuracy					93.1		

are shown over time in Fig. 4.3 (c). As expected, the working routine is the central element within a weekday and only interrupted by some meetings and the lunch break. Likewise, VIL allows to **spot short routine occurrences**, which presents a major advantage in comparison to manual annotations, where these short events can be easily missed. Furthermore, it can **handle time offsets** in the annotations and can provide a more consistent and data-driven labeling. This is important if annotations from multiple test subjects are fused.

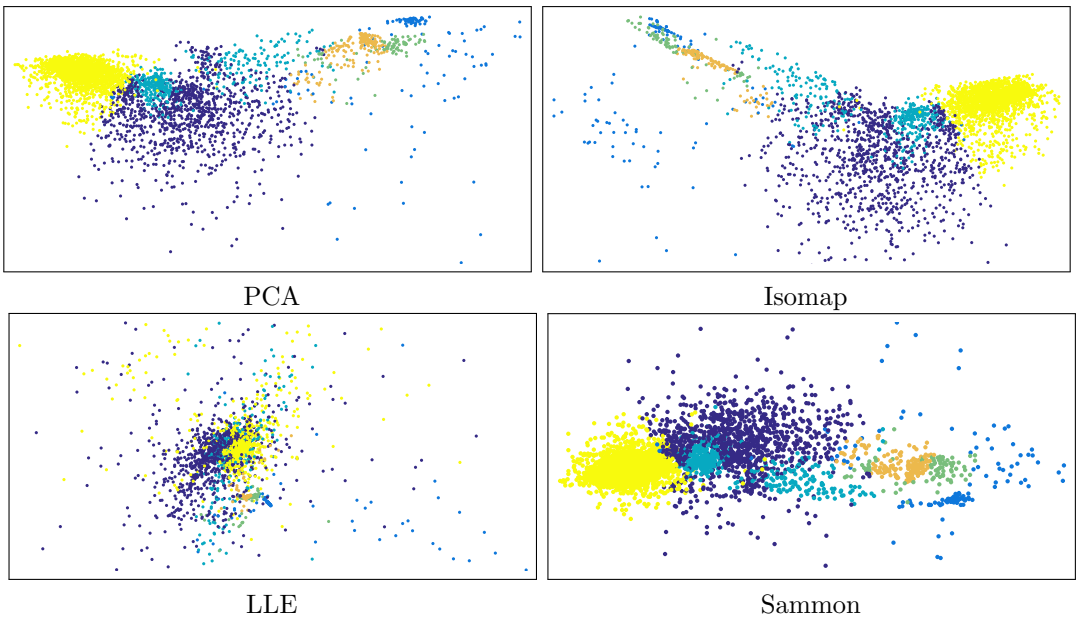
The Huynh data serves as an independent validation set assessing the quality of the VIL. The t-SNE method demonstrated in section 3.4 to produce the best embedding for the VIL procedure, but other embeddings can be provided as well for a different view on the data. The Huynh GT and VIL annotations are compared in 2D plots, over time and by classification measures. Obviously, the difference in Fig. 4.4 (a) is that VIL enforces a smooth label grouping. Thus, for the best performance the classes should be well-grouped in the embedding, but as it is typical in activity recognition not all classes are well separated [44]. Since similar events, such as sitting during work or dinner, cannot be hardly distinguished only by ACC sensors. These wrong predictions over time are evident in Fig. 4.4 by comparing the (b) and (c) time schedule plots. In the classification metrics, this is visible in the low recall and precision values in Table 4.1, but the overall accuracy of 79.5 is reasonable. Therefore, the VIL results on the $\mathbb{D}_{\mathrm{T}}$ set with audio features are expected to be better, which is confirmed by the strong temporal consistency of labels with the time diary annotations. In general, the VIL performance depends how many points you choose, i.e., it will converge to the true Huynh GT, the more points you select. Comparing the VIL results to the DRR outcomes, the fully supervised DRR performance is strongly improved since far more labeled training points are used.

4.2 Comparing the Daily Routine Recognition Rates by Different Input Features

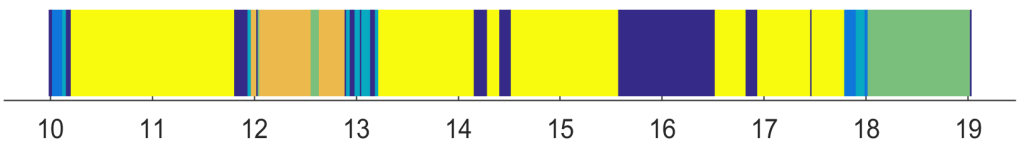
With the found label set on our set $\mathbb{D}_{\mathrm{T}}$, we can recognize the daily routine with various classification techniques and perform a classifier evaluation with different cross-validation schemes to choose the best method. Additionally, the effect of different input data variants is assessed and a comparison to previous work is given on the $\mathbb{D}_{\mathrm{H}}$ set.



(a) VIL annotations as coloring of t-SNE space.

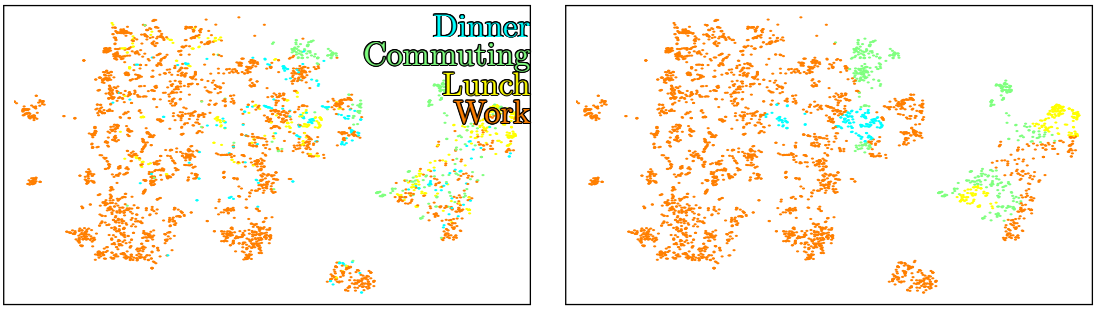


(b) Comparison of embeddings with VIL annotations.



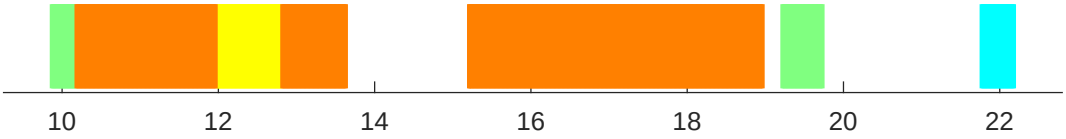
(c) Example day 7 with VIL annotations over daytime [h].

Figure 4.3: The results of VIL annotations are shown in the different embeddings in (a) and (b) plus over time in (c) on the data set $\mathbb{D}_{\mathrm{T}}$. Additionally, the class priors are given in a pie chart and a schematic drawing presents one feature curve of an embedded point.

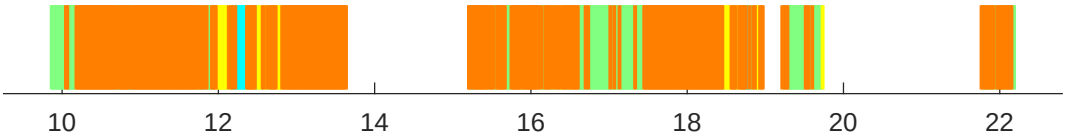


Huynh GT VIL

(a) Huynh GT and VIL annotations on the t-SNE embedding.



(b) Example day with Huynh GT annotations over daytime [h].



(c) Example day with VIL annotations over daytime [h].

Figure 4.4: The Huynh ground truth (GT) and VIL annotations are shown in the t-SNE dimensions and over time with the same coloring.

4.2.1 Classification Techniques

To recognize the daily routine with the learned labels on our set $\mathbb{D}_{\mathrm{T}}$, the input data is the 39-dimensional ACC and audio features. For the selection of a classifier, a comparison between decision tree (DT: medium complexity), multi-layer perceptron (MLP patternnet with 100 neurons), k-nearest neighbor (kNN: weighted), linear discriminant analysis (LDA), random forest (RF: bagged trees), and SVM (coarse Gaussian with a one-vs.-one strategy) is performed. The experiments are done in MATLAB R2018b by the classification learner app with their respective names in brackets except for the MLP model, which is from the deep learning toolbox. The classifiers are explained in detail in section 5.2 except the LDA, which is similar to the PCA projection in section 3.2.1, but it finds a linear projection on which the data of two classes are maximally separated [113]. This is achieved by maximizing the ratio of between-class to within-class scatter matrix. Thus, the best classifier is searched between DT, MLP, kNN, LDA, RF, and SVM.

Afterwards, the chosen classifier is applied to compare the performance of our approach to the previous Huynh results on their data set $\mathbb{D}_{\mathrm{H}}$ with only acceleration features, in [18], and to show the influence of the audio features on our set $\mathbb{D}_{\mathrm{T}}$. Thus, we apply in one variant only the 9-dimensional ACC inputs and in another only the 30-dimensional audio features. We compare these alternatives against the combination with both inputs, ACC and audio. In fourth variant, the 2D t-SNE space is the input.

4.2.2 Classification Evaluation and Experimental Setup

For the classification evaluation, the data set $\mathbb{D}_{\mathrm{T}}$ with the learned labels of VIL stage is randomly split up in v parts. Afterwards, the training is performed on $v-1$ subsets, and the performance is evaluated on the v-th set. This process is repeated for all combinations, which gives a cross-validation (CV) scheme. Finally, the results are averaged over all folds. A more detailed explanation is given in section 5.3.

Ideally, a leave-one-day-out (LODO) CV is applied, since it has not the problem as the leave-one-fold-out (LOFO) scheme that neighboring samples appear in different folds and boost the classification results [114]. Unfortunately, the running exercise and a car ride only occurred on one day in the set $\mathbb{D}_{\mathrm{T}}$. Thus, the results would be degraded by a LODO scheme, because either no training or test data would exist. That is why, the two routines, sport and transport, are removed for the LODO evaluation. But still, the LOFO CV is performed to have detection results for all routines and to keep track of possible biases, both CV schemes are compared. For the Huynh set, we perform a LODO cross-validation. As it is typical in activity recognition, the accuracy, recall, and precision measures are utilized and explained in detail in section 5.3 [44].

4.2.3 Classification Results

For the DRR, comparisons between classifiers, CV schemes, input data, and previous work are performed. On our data set $\mathbb{D}_{\mathrm{T}}$, a comparison of various, cross-validated classifiers in the Table 4.2 states the very good accuracy results of a single decision tree (DT: 92.8%) or even better as an ensemble approach (RF: 95.2%) for LOFO CV. All other learners also perform well in a close margin up to 6.1%. The reason for the ranking is that distances are less meaningful in 39D, which degrades the performance for kNN and kernel SVM. For LDA, finding a linear transformation separating the classes is not totally feasible. The RF is superior to the DT, because ensemble approaches are in statistical advantage over single learners [115]. Thus, the final choice is the best-performing classifier, **random forest**.

As mentioned in [114], the temporal correlation between samples of the same day in different folds boosts the results of LOFO over LODO CV in Table 4.2, but the bias is small for the RF (2.2%) and worse for distance-based classifiers like kNN (6.2%) or kernel SVM (5.0%). The smallest bias has the MLP (0.5%).

To have detection results for all routine activities, the four input data variations – only acceleration, only audio, ACC and audio combined, and the two t-SNE dimensions – are compared in a LOFO CV. The results of RF are presented in Table 4.3. When we only have the acceleration features, the overall accuracy of 80.3% is reasonable. But it has severe problems detecting some classes like transport and TV, since they have low precision values under 50%. That is why, the overall class-averaged precision value has a low value of 63.8% and there is a bigger gap to the class-averaged recall rate of 75.8%. This confusion can be explained, for example, while watching TV and sitting the motion patterns can be like the ones of the majority class (office) work.

In the case of only audio features, the overall accuracy of 92.8% is strongly increased in comparison to only applying ACC inputs. The big enhancement of 12.5% results from the highly improved work, transport, and TV detection, which have the strongest improvements of precision and recall up to 32%. These classes profit from

Table 4.2: Classifier-comparison on $\mathbb{D}_T$ set with acceleration and audio data [%].

Classifier	MLP	SVM	**RF**	kNN	DT	LDA	CV
Accuracy	92.2	90.1	**95.2**	90.3	92.8	89.1	LOFO
Accuracy	91.7	85.1	**93.0**	84.1	90.1	84.6	LODO

Table 4.3: LOFO CV results of **RF** classifier with four input sets: ACC, audio, ACC and audio, and t-SNE (space) [%].

Input	**ACC**		**Audio**		**ACC and Audio**		**t-SNE**	
Routine	R	P	R	P	R(ecall)	P(recision)	R	P
Talk	76.0	75.9	90.5	88.6	89.8	96.5	95.7	96.7
Transport	75.3	44.6	83.8	87.2	95.0	86.9	97.6	92.3
TV	59.9	35.7	74.0	77.6	92.7	74.6	92.6	90.5
Sport	81.5	62.4	83.5	72.4	94.5	81.2	92.9	91.8
Lunch	77.4	70.2	82.0	92.3	89.7	96.9	95.7	95.7
Work	84.4	93.8	98.9	98.7	98.9	99.2	99.5	99.8
Mean	75.8	63.8	85.5	86.1	93.4	89.2	95.6	94.4
Accuracy	80.3		92.8		**95.2**		97.5	

the environmental audio properties, such as low-frequent noise or loudness. Obviously, the talk routine detection performance is raised as expected due to valuable features, such as the own voice activation or speech properties. Therefore, for every class the audio features are better than the ACC ones, i.e., the class-averaged recall and precision values of 85.5% and 86.1% are strongly enhanced.

Adding the audio features to acceleration inputs, the overall accuracy is strongly increased as expected by 14.9% and has a very high value of 95.2%. The class-averaged recall and precision values of 93.4% and 89.2% are also strongly enhanced. In particular, the transport and TV classes have highly increased detection rates up to 42.3% by comparing the two inputs to only using ACC data. The maximal precision value is achieved for the transport class. This means the audio dimension delivers a lot of useful information for the classifier and describes the routines well. By assessing the structure of the RF decision trees, the loudness feature is identified as one of the key separators for different routines. Furthermore, by applying the classifier on the 2D t-SNE space, the results are the best of the four scenarios with an accuracy of 97.5%. Additionally, all recall and precision values are over 90%. This shows the effect of t-SNE mapping, which preserves the characteristics of input space. Additionally, the classifier does not have to cope with the curse of dimensionality as in the 39D space, which results in a higher overall performance.

Consistently, the DRR of our approach on the 36-dimensional Huynh data and annotations is better in a LOFO CV (accuracy of 88.7%) than in LODO (accuracy of 85.8%) due to the existing LOFO bias of neighboring samples in different CV folds [114]. For a fair comparison to Huynh results [18], the time of day feature is added and it highly increases the accuracy to 93.1% because of the very structured routine shown in Fig. 4.4 (b). Their data set has less class transitions and is very repetitive over different weekdays. In contrast to our set, where in another experiment we added the time of day property to the ACC and audio feature vector and it does not improve the classification results in the set $\mathbb{D}_T$. The reason is that the routine classes

are less temporally repetitive. Thus, the time of day feature is a good predictor if someone has a strong repetitive routine and does, e.g., every day the same activities at the same time. For the daily routine recognition, the Table 4.1 states slightly better precision values (Huynh 86.1% to 88.7%) and highly improved recalls (Huynh 67.2% to 84.3%). Thus, our method outperforms the topic model of Huynh.

4.3 Labeling Summary

In this chapter, we used semi-supervised methods to solve the labeling problem of our data set $\mathbb{D}_{\mathrm{T}}$. Based on these annotations, we performed the classification tasks on four input data variants. To validate the extended visual interactive labeling (VIL) method, we applied our approach on the public data set $\mathbb{D}_{\mathrm{H}}$.

Summary

- We demonstrated that the extended VIL works well for consistent data annotations in context of daily routine recognition. It offers the advantages, such as spotting of short routine events or a better handling of time-offsets with a coarse time diary.
- Based on these labels, the classification is performed on the four sets: acceleration (ACC), audio, ACC plus audio combined, and 2D t-SNE embedding. With only the ACC data, only for some routines a good result is achieved.
- We enriched the situational description by including the audio features and produced considerably better results. The classification on the t-SNE space performed the best.
- The VIL method is validated on a publicly available data set and our routine recognition approach outperforms the Huynh results of [18].

Chapter 5

Classifying the Daily Routine Data by Supervised Learning

In this chapter, our goal is the daily routine recognition (DRR) on our data set $\mathbb{D}_7$, and the Huynh set $\mathbb{D}_\mathrm{H}$ based on the classification techniques. The supervised processing scheme is depicted in Fig. 5.1 on our set $\mathbb{D}_7$ with the intention-based classes. We first design an efficient feature representation in section 5.1 and then introduce various classification techniques in section 5.2. This builds the ground for the comprehensive classifier evaluation, where we compare the personalized vs. person-independently trained model performance based on different cross-validation schemes and evaluation metrics explained in section 5.3. Finally, the classification results are discussed in section 5.4. The material introduced is partially taken from our publications [10, 11].

5.1 Feature Extraction and Selection Techniques for Classification

On our data set $\mathbb{D}_7$, we use the features described in section 2.4 for the classification task and an overview of the total feature processing is displayed in Fig. 5.2. To fuse the ACC and audio inputs on the same time grid, the calibrated ACC data is converted to features on an activity primitive level in section 5.1.1. For the acoustic domain, the features are precomputed in the hearing aid. After the creation of the low-level features, the high-level statistical representation is built on a routine activity level in section 5.1.2.

For the Huynh set $\mathbb{D}_\mathrm{H}$, the low-level feature extraction is already performed. Therefore, we use the same high-level representation as for the $\mathbb{D}_7$ set in section 5.1.2.

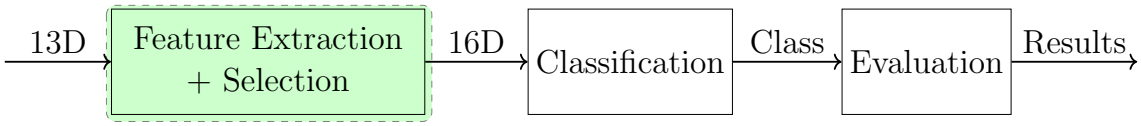


Figure 5.1: The supervised processing scheme for the classification task on our data set $\mathbb{D}_7$ with the intention-based classes, where the dashed block corresponds to the feature section 5.1 shown in Fig. 5.2. The feature representation builds the basis for the comprehensive classifier evaluation.

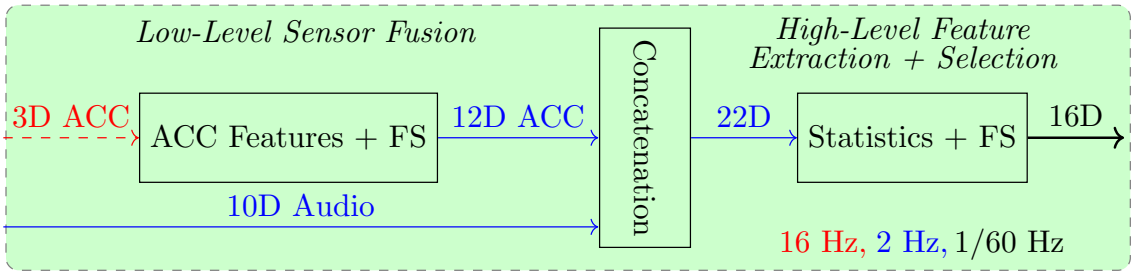


Figure 5.2: The feature extraction and selection scheme consists of two major steps: the low-level sensor fusion in section 5.1.1 and the high-level feature extraction and selection (FS) in section 5.1.2.

5.1.1 Low-Level Sensor Fusion

To design optimal low-level acceleration features for the DRR, we consider a broader set of existing human activity recognition features. These features group into the statistical, time, or frequency domain. A few non-exhaustive examples are given.

The statistical quantities group into positional (such as minimum, mean, median, maximum, or quantile positions) or spread measures (such as range, interquartile range, variance, or standard deviation). The time domain includes, e.g., the integration, auto-correlation, correlation between axes, or mean-crossing rate (MCR).

The frequency-based features are fast Fourier transform bands for 1 to 6 Hz, the spectral energy, spectral entropy, or centroid of spectrum. The mentioned characteristics are helpful to detect activity primitives including walking, various gestures, or head rotations. Thereby, lots of studies showed good recognition rates by finding a suitable set of the introduced accelerometer features for the head and body orientation [42], locomotion [43, 44], conversational gestures [45], and transportation modalities [46].

In the following, we apply the most informative features for our daily routine classes and explain them in more detail. The acceleration values are calibrated and need to be transformed in an informative representation for our target activities. In contrast to the acceleration features, the audio low-level feature extraction was already done in the hearing aid. The 10 precomputed audio features have a rate of 2 Hz and are already explained in section 2.4.2.

To optimally transform the ACC values in a representation, which can differentiate the different daily routine classes, we first consider the signal model and then extract the low-level features. After the calibration of the raw acceleration data with a 16 Hz rate in section 2.4.1, the calibrated triaxial ACC signal $\mathbf{a}_{\text{cal}}$ is ideally only composed of gravitational $\mathbf{g}$, rotational, which splits in radial $\mathbf{a}_{\text{R}}$ and tangential $\mathbf{a}_{\text{T}}$, and linear $\mathbf{a}_{\text{lin}}$ components:

$$\mathbf{a}_{\text{cal}} = \mathbf{g} + \mathbf{a}_{\text{R}} + \mathbf{a}_{\text{T}} + \mathbf{a}_{\text{lin}} \quad \text{in } [g], \tag{5.1}$$

where all quantities are expressed in the hearing aid housing coordinate system and multiples of the earth gravity $g = 9.81 \frac{m}{s^2}$ [87]. The vector $\mathbf{g}$ is only dependent on the HA orientation. If no motion is present, the gravity is directly given corresponding to the head and body orientation.

A taxonomy of possible head rotations and orientations are illustrated in Figure 5.3. The three rotations around the axes are possible. The first one is the yaw

Figure 5.3: Taxonomy of head rotations and orientations adapted from [76] to derive the conversational features.

rotation around the world gravity vector, which is therefore not detectable through the gravity-based orientation estimation and stands for the gesture of saying "no". The pitch rotation corresponds to the conversational gesture of expressing affirmation and the tilting roll angle change shows uncertainty in a chat [116, 117]. Both gestures can be detected by a modification of the gravity vector. Therefore, the orientation is a key identifier to differentiate conversations and further scenes [42]. For example, in our case, the body orientation can distinguish sitting during office work and laying down in a workout. But in case of motion, the **mean** is a typical estimator for the gravity component and the formula is given in Equation (3.1) [118]. We use the low-level mean feature for the estimation of orientation to mainly support the detection of social, listening, or physical class.

The next conversational feature uses the physical property, that the radial and tangential ACC are given by the cross product of sensor position vector $\mathbf{r}$ with angular ACC $\boldsymbol{\alpha}$ and velocity vectors $\boldsymbol{\omega}$ as shown in Equation (2.2). Therefore, both quantities are orthogonal to each other and that is why the **correlation between axes** gives clues about all three possible head rotations. This allows us to detect conversational or listening gestures, such as head shaking or nodding [45]. The Pearson correlation coefficient has a range in the interval [-1,1] and is computed for a segment window of N of two attributes $x_{.1}$ and $x_{.2}$ by:

$$\rho_{\mathbf{x}_{.1}\mathbf{x}_{.2}} = \frac{\sum_{i=1}^{N}(x_{i1} - \mu_{\mathbf{x}_{.1}})(x_{i2} - \mu_{\mathbf{x}_{.2}})}{\sigma_{\mathbf{x}_{.1}}\sigma_{\mathbf{x}_{.2}}}. \tag{5.2}$$

Furthermore, the linear motion component is dominant in comparison to head or body rotations, since the resulting amplitude is far stronger. For this reason, periodic movements, such as walking or jogging, produce a high output and often a variance feature is calculated to distinguish these from being stationary or sitting by the motion strength [44]. The **variance** is useful as well for the detection of transportation modes or sport scenes and is shown in Equation (3.2) [46]. That is why, the variance supports the differentiation of basics to physical or transportation class.

In addition, the **mean crossing rate** (MCR) informs about the motion frequency by counting the number of times the signal crosses the mean value. This is triggered, in particular, by periodic movement. Thus, walking, repeated nodding, or shaking gestures in conversation trigger this measure. The formula is given in

Table 5.1: Summary of applied low-level features for the classification, where the attribute type determines the range of a feature. All ratios except the variance of ACC are between 0 and 1, whereas the intervals have a physical meaning like the mean of ACC between $-2g$ and $2g$.

Input	Methods	Type
Acceleration (12D)	Mean, axes correlation (between two axes)	interval
	mean crossing rate , variance (all for each axis)	ratio
Audio (10D)	Spectral centroid, max. level,	interval
	low- and mid-frequency noise floor,	interval
	own voice activation, absolute correlation, wind,	ratio
	stationarity, 4 Hz modulation, onset detection	ratio

Equation (3.3) and the feature is beneficial to recognize the physical, listening, or social class. Further advanced binaural head rotation features of [84, 119, 120] are not considered, since in our experiments we only use one HA with one acceleration sensor. The reason is that the recorded data of two hearing aids are not synchronous in time, which would be a condition to fuse the ACC data for binaural features.

In total the four measures - mean, axes correlation, variance, and MCR - are extracted of the 3D ACC vector, which gives 12 dimensions. The measures are selected to detect the described behavior of routine classes well. This is done over a sliding window of 1 second with 50 percent overlap, which showed in other studies a good performance for the detection of activity primitives [12]. Conveniently, this matches the 2 Hz rate of 10 precomputed audio features, which are beneficial for all routine classes as explained in section 2.4.2. Therefore, we have a low-level representation of activity and sound primitives on a 2 Hz rate. This allows to detect the various described short-term events and activities. All features are summarized in Table 5.1 with their corresponding attribute type. Thus, we can connect these primitives in the high-level feature extraction in the next section.

5.1.2 High-Level Feature Extraction and Selection

Out of the total 22 audio and ACC low-level features, we can build the high-level routine representation. Therefore, they are segmented in non-overlapping one-minute frames to balance between fast audio (seconds) and slow activity (minutes) changes [41, 92]. This window length already showed a good performance in section 4.2 on Huynh and our set $\mathbb{D}_{\mathrm{T}}$ [9]. Afterwards, the three **statistical** quantities - mean, variance (var), and mean crossing rate - are computed for all features and frames [44]. This summarizes the information about gestures and low-level activities, e.g., the frequency of head rotations and strength of motion, and audio, e.g., changes in loudness levels or own voice activation, on a routine level. Thus, for example, the basics and physical routine can be distinguished by the level of activity. In contrast, the transportation and social routine can be separated by the strength of speech properties or occurrence of low-frequent noise.

To sum up, out of 22 low-level inputs three statistical measures are extracted and 66 high-level features are returned. These attributes are normalized to zero

mean and unit variance by the z-score formula:

$$\tilde{x}_{ij} = \frac{x_{ij} - \mu_{\mathbf{x}_{\cdot j}}}{\sigma_{\mathbf{x}_{\cdot j}}}, \tag{5.3}$$

where the normalization is performed attribute-wise on each sample x_{ij} plus the mean $\mu_{\mathbf{x}_{\cdot j}}$ and standard deviation $\sigma_{\mathbf{x}_{\cdot j}}$ parameters of each attribute are estimated on the training samples. Afterwards, we apply feature selection (FS) methods for the finding an optimal subset of features for the DRR [121]. Therefore, we first preselect a subset of 30 features with the minimal-redundancy-maximal-relevance criterion [122]. Then, we use the computational demanding wrapper-based approach of sequential feature selection (SFS) on the subset in a feasible amount of time [123]. SFS starts with an empty set of features and iteratively adds the feature, which has the strongest increase of the recognition rate until a saturation point is achieved. Then, the final feature representation is found. This is optimized on a leave-one-person-out cross-validation scheme with a random forest classifier. After the SFS, our final feature representation contains $M = 16$ dimensions for the routine recognition.

For the Huynh data set $\mathbb{D}_{\mathrm{H}}$, the low-level feature extraction is already done, since the mean and standard deviation of each feature is calculated at a rate of 2.5 Hz due to storage reasons. Afterwards, we apply the same three statistical quantities in one-minute frames, which gives a 36D space. The time of day completes the 37D feature space and strongly improved the classification rates due to the very repetitive structure of Huynh's working days [9].

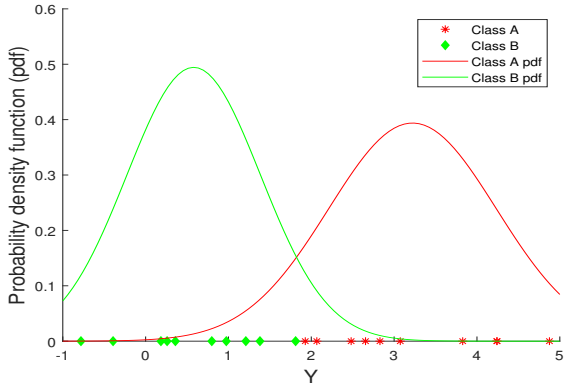
5.2 Classification Techniques

With the found ACC and audio properties in the feature vector $\mathbf{x}$, we classify the routine behavior and environments. Therefore, a set of classifiers is selected for the supervised classification evaluation, which are computationally feasible to use in a HA. The learning principles of all considered fixed offline classifiers are displayed in Fig. 5.4 on the example of a two-class problem in two dimensions. In the following, we introduce them and their specific training procedures. We perform batch learning on the entire training data and apply the fixed classifier model on the unknown test set for the evaluation. The training procedure depends on the chosen learning algorithm, which can be split up into *discriminative* and *generative* techniques. The difference lies in the learning goal. Discriminative approaches directly model the decision boundary between the classes, whereas generative algorithms model the probability distribution of the generated data to deduce the class label based on this distribution. Formally, this is written as the maximum a posteriori (MAP) criterion by the Bayesian decision theory [69]:

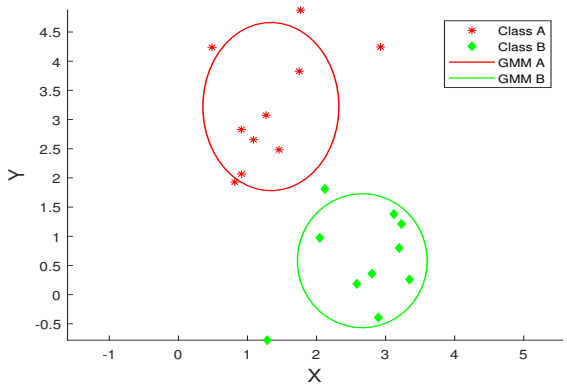
$$\hat{c} = \arg\max_j p(c_j|\mathbf{x}), \tag{5.4}$$

where the decision is made for the class $\hat{c}$ having the highest posterior probability $p(c_j|\mathbf{x})$ of all K classes $\mathbb{C} = \{c_1, \ldots, c_K\}$. It is reformulated by the class likelihood probability $p(\mathbf{x}|c_j)$, class prior probability $p(c_j)$, and data probability $p(\mathbf{x})$ with the Bayes rule:

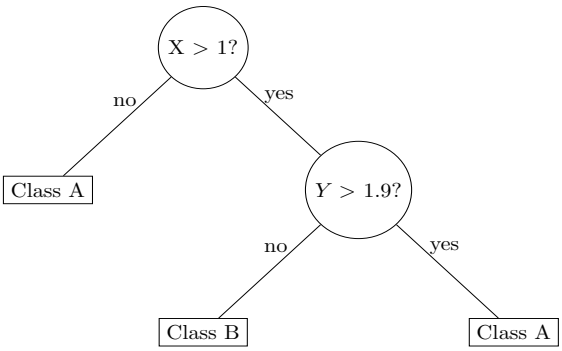
$$\hat{c} = \arg\max_j \frac{p(\mathbf{x}|c_j)p(c_j)}{p(\mathbf{x})}. \tag{5.5}$$



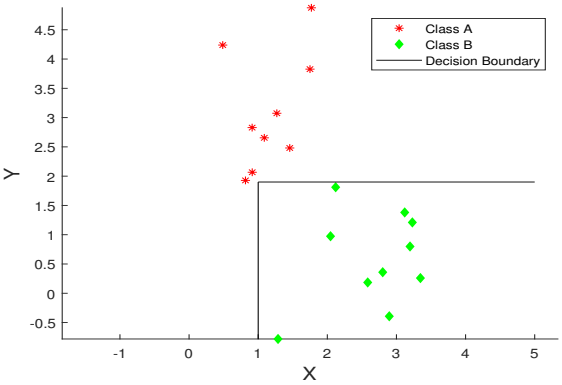
(a) The Naïve Bayes (NB) fits a Gaussian density per class and feature as shown for the Y component.



(b) The Gaussian Mixture Model (GMM) fits a mixture model per class. In the example, one component per class is used.



(c) The Decision Tree (DT) splits the feature space by a Gini criterion.



(d) The resulting DT partitioning of the feature space.

Figure 5.4: The overview and explanation of the main principle for each fixed offline classifier from NB in (a) to DT in (d) on a synthetic binary classification task in two dimensions X and Y.

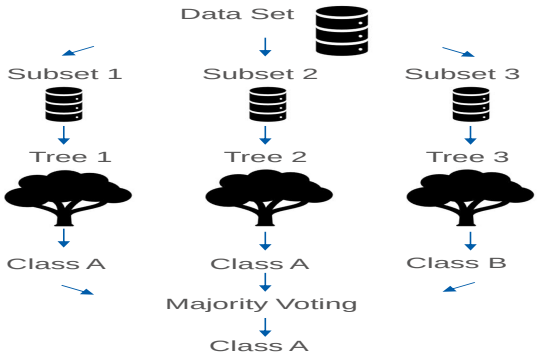
The normalization factor $p(\mathbf{x})$ can be ignored while finding the most probable class or expressed to ensure the posterior probabilities add up to one by the sum rule:

$$p(\mathbf{x}) = \sum_{i=1}^{K} p(\mathbf{x}|c_j)p(c_j). \tag{5.6}$$

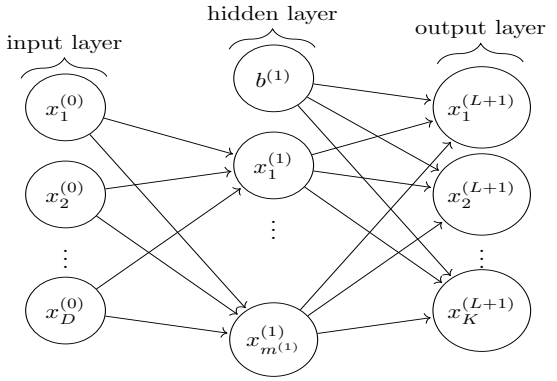
The **Naïve Bayes (NB)** classifier follows the naïve assumption that the $M = 16$ different features of the vector $\mathbf{x}$ are independent of each other given the class [124]. Thus, the class likelihood can be expressed as a product:

$$p(\mathbf{x}|c_j) = \prod_{i=1}^{M} p(x_i|c_j), \tag{5.7}$$

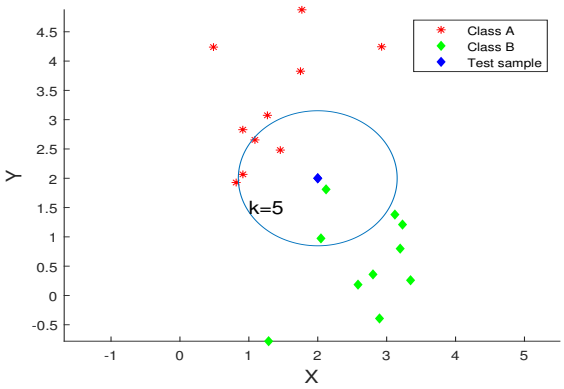
where a one-dimensional Gaussian density models the distribution of $p(x_i|c_j)$. Thus, we need to estimate the mean and variance parameter per feature and class as shown in Fig. 5.4 (a) for a two-class problem on one attribute. The class prior is estimated either with a priori knowledge or from the training data frequency.



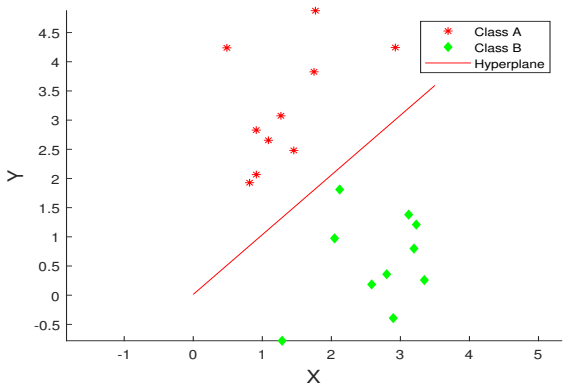
(e) The Random Forest (RF) trains an ensemble of DTs using bootstrapping for the subset selection and a random feature selection per binary split.



(f) The Multi-layer Perceptron (MLP) iteratively trains a non-linear decision boundary by updating weights and biases.



(g) The k-Nearest Neighbor (kNN) predicts the class of a test sample based on the class labels of the nearest neighbors as shown for k=5. In the example, it would be class A.



(h) The Linear Support Vector Machine (SVM) trains a hyperplane to separate the two classes shown.

Figure 5.4: The continued overview and explanation of the main principle for each fixed offline classifier from RF in (e) to SVM in (h) on a synthetic binary classification task in two dimensions X and Y.

In contrast to NB, the **Gaussian mixture model (GMM)** does not assume the independence between the features and models the likelihood by a mixture of a multivariate Gaussian distributions [125]. In the example depicted in Fig. 5.4 (b), two classes A and B are represented by two-dimensional Gaussian distributions with a diagonal covariance matrix. In our case, we fit a mixture model of 8 Gaussian components per class with a diagonal covariance matrix, which is optimized for the tradeoff between the classification performance and computational load. The parameter estimation of mean and covariance is solved with the iterative expectation maximization (EM) algorithm [69]. Again, the MAP criterion is utilized for the classification decision.

The discriminative **decision tree (DT)** represents a partitioning by splitting the feature space in an axis-parallel way by optimal binary decisions of a Gini impurity criterion [126]. An exemplary decision tree with its partitioning is illustrated in Fig. 5.4 (c) and (d), where the two classes are separated by two splits. The Gini impurity criterion of a node T is defined by

$$\mathrm{Gini}(T) = 1 - \sum_{j=1}^{K} [p(c_j|T)]^2, \tag{5.8}$$

which counts the number of times a class occurs at this node. It has its minimum of 0 if all instances fall in one class and its maximum if the instances are equally distributed among the classes. Thus, for each node the feature that yields the highest information gain is selected. The gain is defined by the reduction of parent, T, Gini value minus the weighted impurity of the children nodes T_1 and T_2:

$$\mathrm{Gini}_{\mathrm{gain}} = \mathrm{Gini}(T) - \sum_{j=1}^{2} \frac{N_{T_j}}{N_T} \mathrm{Gini}(T_j), \tag{5.9}$$

where N_T stands for the number of samples in the node T. The last node is the leaf node, in which a decision for a certain class is made.

The **random forest (RF)** builds an ensemble of DTs and takes the majority voting of these trees as the decision [115]. In Fig. 5.4 (e), we displayed the construction principle of the ensemble learner for three trees. In the application, we use 20 decision trees for the RF classifier, which is optimized for achieving a good classification accuracy while being keeping the computational complexity low as possible. To ensure that the deterministic tree induction produces different decision trees, randomization is used. Therefore, for each tree only a subset of all samples is considered by bootstrapping, i.e., every sample has the same chance to be selected. After the bootstrapping, the same decision tree induction scheme like for the DT is performed on each subset with one exception, that for each binary split only a random partition of all features is chosen. In that way, it is ensured that all trees are different and have a diverse knowledge (meaning partitioning of the feature space into class decisions). This is preferable for the generalization abilities of the ensemble classifier. That is why, the RF produces in several classification studies one of the best outcomes [127]. Each DT has the same share in the majority voting and the frequency of votes can be used as a soft score, i.e., it approximates the posterior probability.

The **multi-layer perceptron (MLP)** is a fully-connected feed-forward neural network with one hidden layer $L = 1$ shown in Fig. 5.4 (f) [128]. In case of multiple hidden layers, it is called a deep neural network. The MLP consists of a number of neurons, $m = 100$, per layer. The hidden layer performs a weighting of the inputs $\mathbf{x}^{(0)}$ plus adds a bias term b that is fed in a non-linear activation function f: $x_i^{(1)} = f(\mathbf{w}_i^{(1)} \cdot \mathbf{x}^{(0)} + b_i^{(1)})$. We use the rectified linear unit, ReLU, function that returns the maximum of weighted inputs or zero: $f(x) = \max(x, 0)$. The network output is generated by the so-called softmax function that converts the output $x_i^{(L+1)} = \mathbf{w}_i^{(L)} \cdot \mathbf{x}^{(L)} + b_i^{(L)}$ of $(L+1)$th layer to a soft score representing the posterior probability:

$$\hat{p}(c_j|\mathbf{x}) = \frac{\exp(x_j^{(L+1)})}{\sum_{j=1}^{K} \exp(x_j^{(L+1)})}. \tag{5.10}$$

For the training the network, the weights and biases are randomly initialized and then the iterative optimization adapts them. Thereby, the softmax output of the forward pass is compared to actual posterior probability by the cross-entropy loss. These results are fed back in the backward pass based on the derivatives with the

Adam optimizer and a L2 regularization term to update the weights and biases [129]. Thus, the MLP iteratively trains a non-linear decision boundary with one hidden layer, where 100 hidden neurons in this layer are a good tradeoff between complexity and performance.

The **k-nearest neighbor (kNN)** predicts the class of the nearest neighbors by finding the smallest Euclidean distance between the training examples and the test sample. Thus, it follows the idea that close points within the feature space should have the same label, since they share the same characteristics. Ideally, the different classes are well separated in the feature space. To do so, kNN stores the training data to compute the distance measure shown in Equation (3.5). Afterwards, the k closest training examples are found, and the unknown class label is given by the majority voting of all nearest neighbors. Since every neighbor has the same influence (uniform voting) on the outcome, it can happen that more distant neighbors can produce misclassified instances, in particular, for a larger value of k. We use k=5 neighbors to mitigate this effect. In the example depicted in Fig. 5.4 (g), this situation happens, since the majority votes is for class A. Another way to cope with this situation is to apply a weighted voting scheme, where the inverse distance defines the voting share, and the highest voted class is selected. Therefore, closer points have more influence, i.e., in the example in Fig. 5.4 (g) the weighted voting leads to a decision for class B because of the very close neighbor. We tested both voting schemes plus a number of values for k and the best combination is k=5 with the uniform voting in terms of efficiency and classification outcome.

The **support vector machine (SVM)** is a binary classifier, which applies a linear hyperplane as the decision boundary to separate the classes as shown in Fig. 5.4 (h) [130]. The hyperplane is characterized similar to a MLP neuron by

$$\mathbf{w}^{\mathrm{T}} \cdot \mathbf{x} + b = 0. \tag{5.11}$$

The question is how to find an optimal hyperplane that generalizes well for unseen class instances. It is assumed, that this hyperplane has a maximal margin to the class instances, which is measured by the distance between the hyperplane and the so-called support vectors. This works well if the problem is linearly separable like in our example in Fig. 5.4 (h) and the maximal margin implies a small generalization error. In non-separable cases, it is allowed that some instances are misclassified by introducing slack variables. For a multi-class problem, the binary SVM classifier is applied in a one-vs-all or one-vs.-one classification strategy, where the multi-class problem is transformed in multiple binary decision problems. For a non-linear SVM classifier, the so-called kernel trick is applied, where the data is mapped to a high-dimensional feature space, in which the problem should be linearly solvable. A common choice is the Gaussian radial basis function kernel, but the non-linear SVM has a strong increase in computational complexity. Therefore, we apply the more efficient linear SVM with a one-vs-all classification.

In this chapter, we compare the daily routine recognition performance between the following previously classifiers with the applied parameters:

- decision tree,
- random forest ensemble of 20 decision trees,
- multi-layer perceptron with a hidden layer consisting of 100 neurons,

(a) The leave-one-person-out (LOPO) scheme

(b) The leave-one-fold-out (LOFO) scheme

(c) The leave-one-day-out (LODO) scheme

Figure 5.5: Offline evaluation schemes groups the data set by person, random fold, or day.

- k-nearest neighbor with k=5,
- Gaussian mixture model with a mixture model of 8 components per class and a diagonal covariance matrix,
- Naïve Bayes with one Gaussian likelihood density per feature and class, and
- linear SVM with a one-vs-all classification.

5.3 Classification Evaluation and Experimental Setup

Evaluating the offline classification, the data set is split up in v parts as shown in Fig. 5.5. Afterwards on $v-1$ (black) subsets, the training is performed and on the unseen v-th (red) set the predictions are made. This process is repeated for all combinations, which gives a **cross-validation** (CV) scheme. Finally, the metrics are computed over all subsets.

The three applied CV schemes, **leave-one-person-out** (LOPO), **leave-one-fold-out** (LOFO), and **leave-one-day-out** (LODO), differ how they group the data set. Hence, LOPO splits person-wise, LOFO groups them in five random subsets of the same size and LODO makes one group per day for each subject. The LOPO scheme assesses the recognition rate of a person-independent model. On contrary to the LODO grouping, that is applied for a personalized training. As suggested in [114] for the LOFO scheme, there might exist a possible bias. Due to random split of the temporal data, neighboring samples can appear in different folds and are likely to be highly correlated. This results in over-optimistic recognition rates. We test if this bias is also present for high-level activities.

As it is typical in activity recognition, the measures,

- the **confusion matrix** summarized by four events: true positive (TP), true negative (TN), false positive (FP), and false negative (FN),
- **accuracy** $A = \frac{TP+TN}{TP+TN+FP+FN}$, and
- F_1**-measure** as harmonic mean of recall $\frac{TP}{TP+FN}$ and precision $\frac{TP}{TP+FP}$,

are applied [44]. We use the class-averaged F_1-measure and not the weighted version, since the data set has a strong class imbalance shown in Fig. 2.4 and the overall weighted performance would be dominated by the majority classes. That is why, the reported F_1-rates are expected to be lower than the weighted version, since the class averaged F_1-measure is independent of the class distribution [44]. Therefore, the scores reflect better the minority class performances than the accuracy criterion, which is affected by the class-imbalance and dominated by the majority class scores. Thus, we can judge from the ratio between the two metrics how well a classifier is doing in the overall performance as well as for the minority classes. This means if a classifier has a stronger gap between the A and F_1-measure, the model does not recognize the minority classes well. However, if both metrics are on a par, all classes are equally well detected. We use the confusion matrix for a detailed picture of misclassified routine activities with the best-performing algorithms.

We use a **statistical significance test** to show that the performance differences between two classifiers are not by chance but are statistically significant. The test assumes that the CV scores are drawn from the same distribution and if we can reject this null hypothesis, it means the differences in the scores are statistically significant. We apply the non-parametric Wilcoxon's signed-rank test that does not make a distribution assumption on the results of the CV folds [131]. The hypothesis test compares the cross-validation results of two classifiers. Therefore, each result per fold is considered as a trial and the performance differences of the two classifiers are computed. The absolute values of these differences are ranked and for each classifier these ranks are summed, on which a classifier won. If the compared lower sum is below a critical value, the null hypothesis is rejected. We apply a confidence level of 5% that the performance of these classifiers is no different. Thus, one classifier is significantly better than the other.

5.4 Classification Results

In the offline experiments on the $\mathbb{D}_7$ set, we analyze the A and F_1 performance of all classifiers based on three cross-validation schemes: LOPO, LOFO, and LODO. Thereby, we assess the person-dependent and -independent classification rate and look for a possible bias between the LOFO and LOPO CV schemes. The comparison of various classifiers in the Table 5.2 states the very good F_1 results of the MLP network (78.7%) or the ensemble approach RF (79.2%) for the LOPO CV. To assess the statistical significance of these results, the Wilcoxon hypothesis test is performed. It shows that these two classifiers are significantly better than all others, but not to each other. The most learners also perform well in a margin about 8% worse except the density-based classifier NB, which is more negatively affected by the class imbalance [132]. The kNN is on the third place with a medium distance up to 4% to the MLP and RF, but it is followed in the range of 4-5% by the GMM and SVM. Further reasons for the ranking are that the linear decision boundary of the SVM is not complex enough to separate all classes. Thus, the MLP with a non-linear boundary distinguishes better between the classes. The kNN classifier learns by example and moderately generalize well over unseen data of different users. The RF is superior to the DT, because ensemble approaches are in statistical advantage over single learners [115].

To analyze the **class imbalance** effects, we compare the F_1 results in Table

Table 5.2: Results of offline classifier $\mathbf{F_1}$ performance [%].

CV	SVM	NB	MLP	kNN	GMM	DT	**RF**
LOPO	70.1	65.9	78.7	75.2	71.2	69.0	**79.2**
LOFO	72.9	66.3	83.2	81.9	74.0	76.5	**83.9**
LODO	74.6	69.1	78.2	77.0	73.2	71.0	**79.5**

Table 5.3: Results of offline classifier **accuracy** performance [%].

CV	SVM	NB	MLP	kNN	GMM	DT	**RF**
LOPO	78.8	73.2	83.2	80.7	77.1	74.9	**83.8**
LOFO	80.7	73.4	86.4	85.3	79.2	80.6	**87.0**
LODO	81.2	74.6	83.3	82.0	78.7	77.1	**84.2**

5.2 to the A rates in Table 5.3 for the LOPO cross-validation. The MLP and RF classifiers have the smallest difference between F-measure and accuracy rate of 4.5% and 4.6%, i.e., they are less affected by the class imbalance. In contrast the NB and SVM learners, they have the largest difference of 7.3% and 8.7%, whereas kNN, GMM, and DT are in-between with a difference of about 6%. The kNN is on the third place in the LOPO accuracy ranking with a medium distance of 3% to the best performers MLP and RF, but it is closely followed by the GMM and SVM.

As mentioned in [114], the **temporal correlation bias** between consecutive samples in different folds boosts the F_1 results of LOFO over LOPO CV in Table 5.2 in the interval of 0.4% to 7.5%, where the DT is affected the most. This bias is smaller for the MLP network (4.5%) than for the ensemble method RF (4.7%), but worse for instance-based classifiers like kNN (6.7%). The parametric density estimation of NB (0.4%) and GMM (2.8%) is less affected by the temporal sample correlation, since the parameter updates are aggregated over the whole training data. This is also the case for the SVM parameterization of linear hyperplane and further non-tested classifiers, which follow the same learning principle.

Furthermore, we compare the LOPO and LODO CV to assess the **personalization** effect of the classifiers. The biggest F_1 and A improvement of 4.5% and 2.4% has the SVM classifier and it is closely followed by the NB and DT with an improved F_1 rate of 3.2% and 2%. The DT has a similar accuracy enhancement of about 2% like the SVM, whereas kNN, NB, and GMM closely follow by an improvement of about 1.5%. The RF has the smallest F_1 enhancement of 0.3%, whereas the MLP slightly degrades its F_1 performance, but the overall accuracy slightly improves. Thus, the MLP improves the majority class performance while decreasing the rate on the minority classes. The classifier ranking remains similar to the LOPO case with smaller deviations. The Wilcoxon hypothesis test demonstrates that the RF classifier is significantly better than all others in the personalized LODO cross-validation. Therefore, we confirm the previous literature results [43, 66], where the person-dependent model with (LODO) CV scheme performs better than the independent one. This holds not only for low-level activities, but it is also valid for the high-level routine activities.

The **detailed results of best performer RF** are presented by the confusion matrix in Fig. 5.6, where the class-wise recall is shown in the rows. Obviously, three

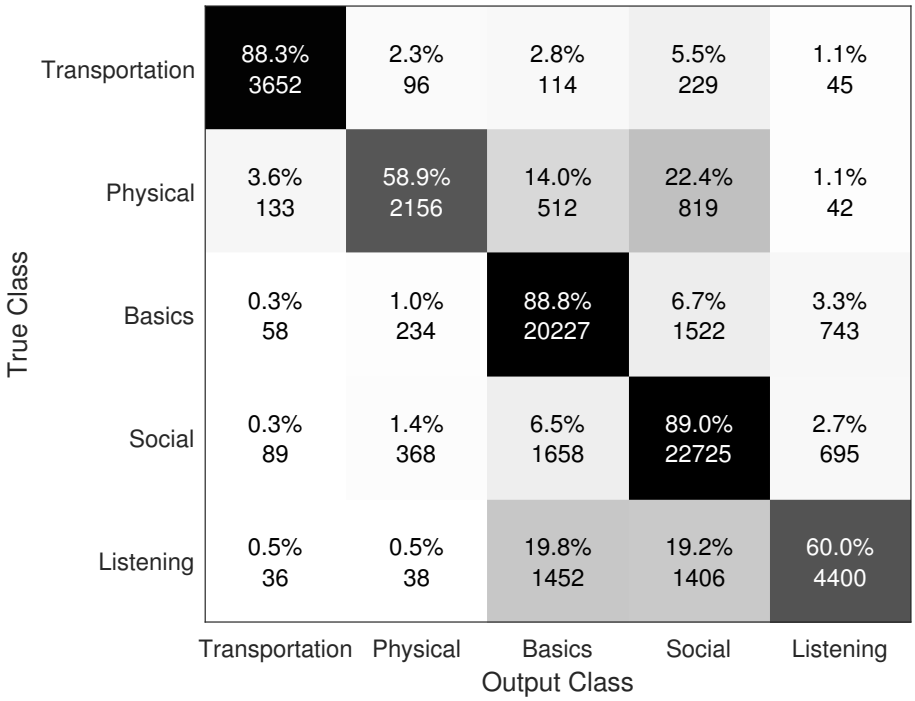


Figure 5.6: Confusion matrix of LOPO with RF.

of five classes are very well detected with over 88% of recall and they contribute as majority to the high overall accuracy of 83.8%. The biggest confusion stems from the listening class with basics (19.8%) and social (19.2%). This makes sense due to the close relation between listening and social, where class transitions happen quite often. Likewise, the difference between listening and basics is mainly detected due to different audio characteristics, but for some situations they could be similar. For example, there is a background conversation, and the subject does not want to follow it. Thus, a possible source for the classifier confusion stems from this intention-based scenery. Furthermore, the bigger mismatch of 22.4% between physical and social happens since both classes could also be simultaneously active and then the user's intention decides. Here, specialized acceleration conversation or movement features could deduce the motion behavior and the situational intention more precisely.

Additionally, we analyzed if less transmitted data due to the varying Bluetooth data transmission, i.e., the missing feature problem, correlates with wrongly predicted samples. Therefore, a histogram with the number of transmitted samples per segment window given the correct or wrong prediction outcome states that both distributions are nearly identical. Thus, the daily routine recognition on our statistical features is robust to the missing feature problem.

5.5 Classification Summary

To summarize our findings in this chapter, we recognized the daily routine on our data set $\mathbb{D}_7$ based on offline classification approaches. Therefore, we designed a robust statistical feature representation and explained various classification techniques. To evaluate the model performances, we introduced three cross-validation schemes and compared the recognition rates on suitable metrics.

Summary

- For the offline recognition on high-level activities, we confirm that the person-dependent model is superior to person-independent classifier.
- We showed that the temporal sample correlation in different folds returns over-optimistic results in the leave-one-fold-out cross-validation.
- We further demonstrated that the multi-layer perceptron and random forest yielded the best F-measure of 78.7% and 79.2%.
- The remaining misclassified samples require a tailored motion representation to distinguish the intended behavior more precisely.

Chapter 6

Improving the Daily Routine Classification by Modeling the Temporal Behavior

In this chapter, our goal is to improve the daily routine recognition (DRR) on our data set $\mathbb{D}_7$, and the Huynh set $\mathbb{D}_\mathrm{H}$ by modeling the temporal behavior with online and sequence learning methods. We use the efficient feature representation of section 5.1 as a basis for the various online and sequence classification techniques. The online methods in section 6.1 adapt the classification models over time based on the incoming data stream, i.e., an offline trained person-independent start model is personalized on daily updates that should enhance the recognition rates. This should imitate the behavior of a real system application. In contrast to online methods, the sequence classifiers in section 6.2 take into account the order of the routine data to model the relationships between neighboring samples, which should improve the temporal stability of the classification predictions and the performance. For both scenarios, we perform a comprehensive classifier evaluation, where we compare the model performances. Finally, the classification results are discussed and compared to our expected improvement. The material introduced in this chapter is partially taken from our publications [10, 11].

6.1 Adapting the Classification Models by Online Learning

To adapt the daily routine classifiers, we introduce various online learning techniques in section 6.1.1. The evaluation and experimental setup are explained in section 6.1.2, which determines the online classification results in section 6.1.3. The proposed procedure is only possible on the $\mathbb{D}_7$ set due to the multiple subjects are needed in contrast to $\mathbb{D}_\mathrm{T}$ and $\mathbb{D}_\mathrm{H}$ sets.

6.1.1 Online Learning Techniques

For the online classification, the initial model is trained on all known subjects and then personalized on the unknown test person P by daily online updates. The initial training is performed in a leave-one-person-out (LOPO) manner and thus, it is called

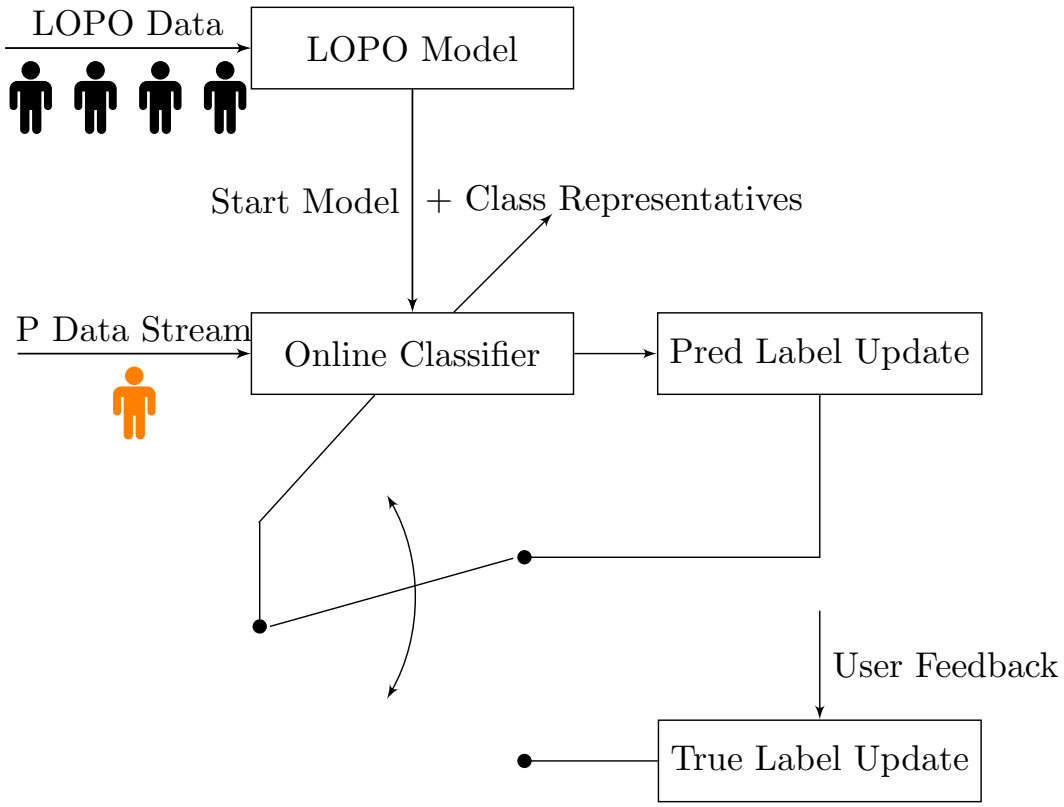
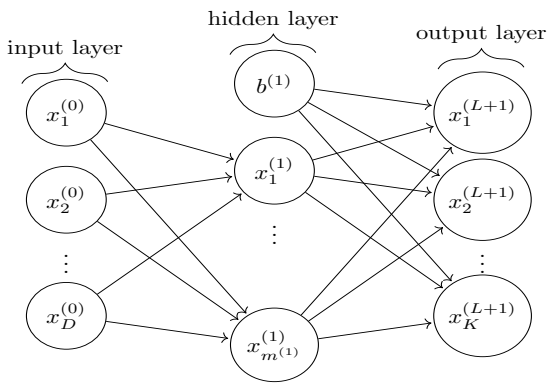


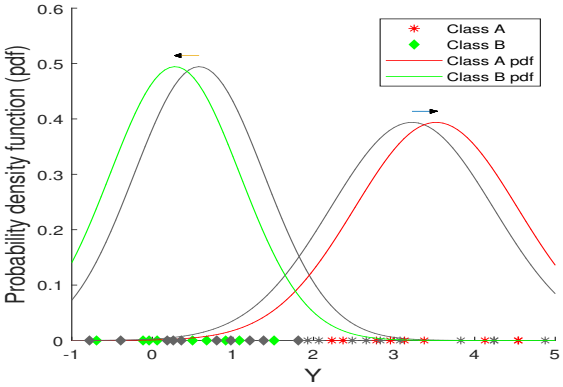
Figure 6.1: The online classifier adaptation starts with a person-independent LOPO model and personalizes its model based on the true or predicted label updates of the incoming personal data stream of test person P. The true labels need a user feedback, where the predicted labels are the own model predictions. The LOPO data refer to the training data of known subjects.

LOPO model. Then, the online learning methods adapt the classifiers, and the principles of this process are shown in Fig. 6.1. Thereby, the online personalization updates the classifiers based on the data of the new day in two ways either with the true labels ("**true update**") or the own predictions ("**pred update**"). For the "true update", we assume the user annotates the new data, for example, in a smartphone, whereas for the "pred update", the current model predicts the labels of the new day and uses these for the training. Hence, we analyze if the classifier can self-improve over time without a necessary user-feedback.

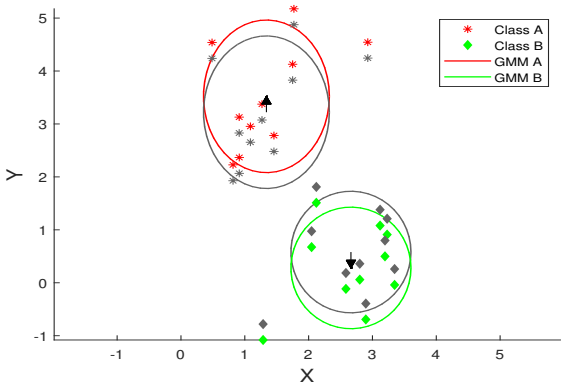
Unlike in the offline phase in section 5.2 with batch training, all classifiers are trained in data chunks, where the first one consists of all known LOPO subjects, and the remaining ones are the incoming data of the new test person P on a daily basis. The one-day adaptation interval is chosen, since the daily routine activities are conducted over a time frame of minutes to hours. In contrast to the daily adaptation, a shorter update interval would have a smaller number of existing instances of different classes. Thus, we ensure a broader data variability of multiple existing classes for each daily adaptation, which should ease the model generalization. In doing so, we imitate the behavior of a mobile system in real life, where the updates take place, for example, in a smartphone and only the adapted parameters are transferred to the HA. This is more computational- and energy-efficient than updates in shorter intervals.



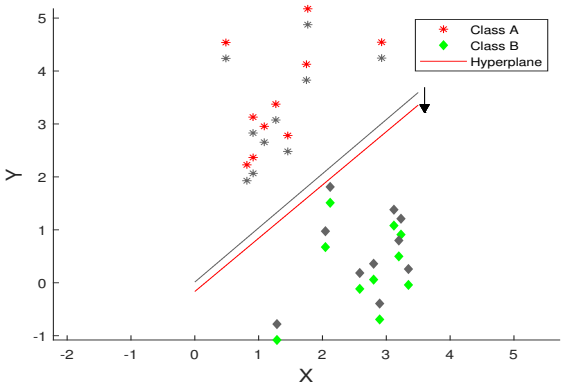
(a) MLP performs incremental forward and backward passes to update the weights and biases on the incoming data.



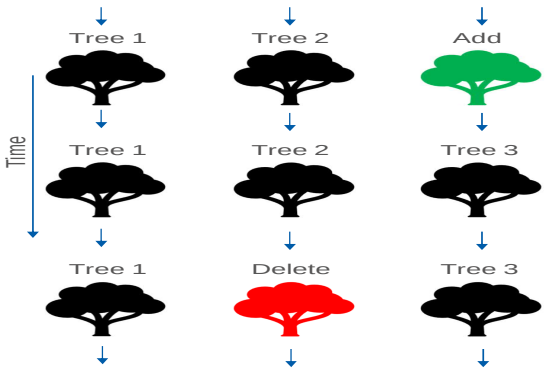
(b) NB updates the Gaussian distribution parameters, mean and variance, for each feature and class.



(c) GMM updates the mean component of the Gaussian mixtures.



(d) Linear SVM shifts the hyperplane to adapt its margin.



(e) RF adds and deletes ensemble trees to update the model over time.

Figure 6.2: The overview and explanation of the main principle for each online classifier from MLP in (a) to RF in (e) on a synthetic binary classification task in two dimensions X and Y. The old data is marked in gray, whereas the newly incoming data chunks are in colors and are used to adapt the classifiers.

Since a recurring drift may occur, meaning stronger data distribution changes due to different performed activities, e.g., between weekends and working days, we need to ensure that the classifier does not drastically change or forgets too much knowledge. This is achieved by adding representative feature points of the offline LOPO data for each class to incoming personal data for each training update. To find appropriate representative samples for each class, we use the k-means clustering algorithm introduced in section 3.2.1, since it equally distributes the representative samples over the space of each class. We optimized the number of centers to keep the storage requirements low while preserving as much as possible the class knowledge. 100 centers per class showed the best trade-off. An additional benefit is that all classes are present for each of the daily updates.

Furthermore, the classes are differently imbalanced across subjects as shown in Fig. 2.4 and the online personalization addresses this issue by adapting the classification models. Additionally, the representative features for each class play an important role in an imbalanced classification problem to separate the classes well [133]. That is why, we choose the features calculated in section 5.1 in the data set $\mathbb{D}_7$ to be informative and discriminative for these classes to optimally separate them.

Online Update Mechanisms

The learning principles of online classifiers are displayed in Fig. 6.2. Therefore, we use the so-called partial or incremental fits of the MLP, NB, and SVM classifiers [63, 134], i.e., the parameters of the neurons, Gaussian densities, and hyperplanes are iteratively updated. Therefore, the **MLP** performs the same online forward and backward passes on the personal incoming data of P as explained in section 5.2 for the offline classification and is shown in Fig. 6.2 (a). Thereby, the weights and biases of each node are updated.

The **NB** classifier updates the model parameters, mean and variance, of each Gaussian likelihood per feature and class plus the prior distribution [135]. If we have N_1 samples of one feature $x_1, x_2, \ldots, x_{N_1}$ for the initial fitting and N_2 samples of the same feature $x_{N_1+1}, x_{N_1+2}, \ldots, x_N$ for the parameter update, where in total we have $N = N_1 + N_2$ samples. Then, the two means, μ_{1,N_1} and $\mu_{N_1+1,N}$, are given by

$$\mu_{1,N_1} = \frac{1}{N_1} \sum_{i=1}^{N_1} x_i \quad \text{and} \quad \mu_{N_1+1,N} = \frac{1}{N_2} \sum_{i=N_1+1}^{N} x_i, \tag{6.1}$$

which allows to compute the mean of all samples:

$$\mu_{1,N} = \frac{N_1 \mu_{1,N_1} + N_2 \mu_{N_1+1,N}}{N}. \tag{6.2}$$

To calculate the variance σ^2, we introduce the sum of squares, S_{1,N_1} and $S_{N_1+1,N}$, for both subsets

$$S_{1,N_1} = \sum_{i=1}^{N_1} (x_i - \mu_{1,N_1})^2 \quad \text{and} \quad S_{N_1+1,N} = \sum_{i=N_1+1}^{N} (x_i - \mu_{N_1+1,N})^2 \tag{6.3}$$

and this allows to define the total sum of squares $S_{1,N}$:

$$S_{1,N} = S_{1,N_1} + S_{N_1+1,N} + \frac{N_1 N_2}{(N_1 + N_2)} (\mu_{1,N_1} - \mu_{N_1+1,N})^2. \tag{6.4}$$

To derive the variance parameter for each class and feature, we need to normalize the sum of squares by the number of samples N for a biased estimation or reduce it by one to $N-1$ for an unbiased estimate. In Fig. 6.2 (b) an example classification of two classes A and B is shown, where the gray probability density functions (pdfs) and class samples represent the results of the initial LOPO training. After the online parameter update on the colored new samples, the pdfs are changed and slightly shifted to the sides, since in the example only the mean component changed.

For the **GMM**, the adaptation of the mean parameter is used, since this is very efficient and changing the covariance showed only a minor improvement [136]. Assuming we have N_1 samples for the initial model training of one class and N_2 feature vectors $\mathbf{x}_{N_1+1}, \mathbf{x}_{N_1+2} \ldots, \mathbf{x}_N$ for the update. For each of the $m \in 1, \ldots, L$ Gaussian mean vectors, we determine the weighting or soft allocation factor $\lambda_i(m)$ in the interval [0,1] for each feature vector $\mathbf{x}_i$:

$$\lambda_i(m) = \frac{w_m \mathcal{N}(\mathbf{x}_i|\mu_m, \mathbf{\Sigma}_m)}{\sum_{j=1}^{L} w_j \mathcal{N}(\mathbf{x}_i|\mu_j, \mathbf{\Sigma}_j)}. \tag{6.5}$$

To update the mean vector $\boldsymbol{\mu}_m^{(new)}$ of the m Gaussian components, the soft allocation factor determines the influence of each feature vector on the new mean component:

$$\boldsymbol{\mu}_m^{(new)} = \frac{N_1 \boldsymbol{\mu}_m^{(old)} + \sum_{i=N_1+1}^{N} \mathbf{x}_i \lambda_i(m)}{N_1 + \sum_{i=N_1+1}^{N} \lambda_i(m)}. \tag{6.6}$$

The linear **SVM** updates the hyperplane parameters of Equation (5.11) on the incoming data stream based on a hinge loss and stochastic gradient descent scheme, which shifts the hyperplane in space [130, 137]. The stochastic gradient descent is defined by updating the weights

$$\mathbf{w}^{(new)} = \mathbf{w}^{(old)} - \eta \frac{\partial E(\mathbf{w})}{\partial \mathbf{w}}, \tag{6.7}$$

where the new weights are the sum of the old weights and the negative gradient of the error term times a learning rate factor η. An example is displayed in Fig. 6.2 (e), where the two classes move away from each other, and the hyperplane adapts this position to keep its margin.

In our own ensemble method, we combine the known Learn++ and online random forest approach of [65, 66] by implementing a new extended **RF** technique with an online adaptation mechanism, which adds and deletes ensemble trees over time as shown in Fig. 6.2 (e). This adapts the model to a possibly instationary behavior or interpersonal differences. The online RF is constructed as follows. The initial model is trained with 10 trees and for each daily update, we train two additional RF trees on the incoming data chunks. Thus, the *adding mechanism* includes new knowledge in the ensemble. This adapts the classifier to recurring daily behavior, like in weekends, by the daily training of new trees while preserving the old LOPO knowledge, since we have more trees trained on the initial LOPO data. The online training of new trees is performed in the same manner as for the offline RF trees explained in section 5.2. The fixed baseline ensembles train 20 trees, which already showed in section 5.4 a good classification performance. The *forgetting mechanism*

checks the individual accuracy A_i of all T ensemble trees $i \in 1, 2, \cdots, T$ on the incoming data and assumes the rates are Gaussian distributed. The mean μ_A and standard deviation σ_A of all individual accuracy rates are computed. We apply a modified z-score outlier criterion determining if a tree has a significantly weaker performance than all other trees [138]:

$$A_i - \mu_A < -2\sigma_A. \tag{6.8}$$

If a tree is worse than minus two times the standard deviation of mean performance, this tree is deleted, since it has worse a performance than approx. 98% of all other trees. Therefore, the old non-appropriate knowledge is deleted, and the relative performance of the ensemble is increased. The online RF has a variable number of decision trees and starts with 10 of them. At the beginning, the number of trees usually grows linearly since new knowledge about the previously unknown test person is included in the ensemble classifier by the adding mechanism. In a later stage, the forgetting mechanism removes more and more trees with outdated knowledge based on the relative performance criterion in Equation (6.8). The final ensemble had about 30 to 40 decision trees.

Setup of the Online Investigations

Furthermore, the online algorithms are compared to four baselines, which apply the offline training procedure of section 5.2, fit(input data), and one variant uses the online adaptation procedure of this section, adapt(input data). The training is performed on the different input data variants, i.e., the varying training procedures use the data of the known training subjects (LOPO data, 👨👨👨👨) or the training data of the test person P (P data, 🕘$_1$ 🕘$_2$) or both as shown in Fig. 6.3:

- the person-independent, initially-fitted LOPO model is called: "**fit(LOPO)**",
- the person-dependent model that is only trained on the test person's data: "**fit(P)**",
- the combination of both personal and LOPO data in one model fit: "**fit(LOPO+P)**", and
- the person-independent classifier that is fine-tuned by the personal data: "**fit(LOPO)+adapt(P)**".

The fit(LOPO) model is the lower bound for the online updates, since the initially-fitted LOPO model should be improved by the online training. The other three baselines denote the upper bounds for an algorithm, which should be the maximal possible improvement for the incremental training. We have three procedures to find the upper bounds and we compare them to figure out, which one is optimal learning procedure for a specific classifier. Thus, if a learner solely performs better on the personal data P or can leverage a higher predictive power from a greater amount of training data by adding the personal-independent LOPO data. Thereby, we cross-compare if it is more beneficial to train the classifier in one model fit, fit(LOPO+P), or fine-tune the initial LOPO model on the personal data, fit(LOPO)+adapt(P).

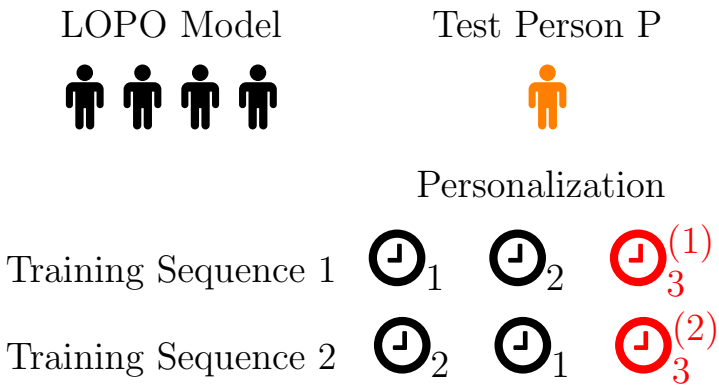


Figure 6.3: Online evaluation scheme is a combination of a leave-one-person-out (LOPO) and leave-one-day-out (LODO) cross-validation, i.e., first the scheme iterates over the subjects and then over the days of the current test person P. This means the model is initially trained on the person-independent LOPO data and then personalized in daily updates on the data of the test person P. The training data of LOPO subjects and test person P are marked in black and test days of person P are in red. Two repetitions of training sequences 1 and 2 with the same test day are shown. The cross-validation repeats then all combinations for the test days and subjects.

All experiments are done in MATLAB R2019b in conjunction with classifiers from the Python library scikit-learn 0.22.2 [134]. For all methods, the made changes from the default parameters are explicitly mentioned. We implemented the online RF and GMM classifiers in MATLAB with the fitting functionality of Python for the RF trees and GMM probability distribution.

6.1.2 Online Learning Evaluation and Experimental Setup

Evaluating the online classification, we use a combination of LOPO and LODO scheme shown in Fig. 6.3. First, the LOPO model is initially fitted on $v-1$ known subjects and the unseen v-th subject is test person P, on which the online personalization is performed. Therefore, the daily training updates are done on the training days $\textcircled{1}$, $\textcircled{2}$ and the recognition rate is reported on the fixed test day $\textcircled{3}$ for each step. To further analyze the influence of training day order, we test all permutations of the training sequences as seen in Fig. 6.3. For the example of three days, the two possible training sequences $\textcircled{1}$, $\textcircled{2}$ and $\textcircled{2}$, $\textcircled{1}$ are displayed with a fixed test day $\textcircled{3}$. These training days $\textcircled{1}$ and $\textcircled{2}$ are the personal data of P, which is used to train three baselines. Of course, the performance metric is always estimated on the fixed unseen test day $\textcircled{3}$ and is averaged over all repetitions ($\textcircled{3}^{(1)}$,$\textcircled{3}^{(2)}$). Again, as in the LODO scheme, all day permutations are simulated, and the results averaged over all combinations. Afterwards, the online simulation is repeated for all subjects and the final performance is averaged over all outcomes. Therefore, we summarize the calculations for the final online results in Table 6.1 and 6.2 plus show the computations on the example of Fig. 6.3:

1. average performance over the training sequences ($\textcircled{3}^{(1)}$,$\textcircled{3}^{(2)}$),

2. average performance over the test day combinations ($\textcircled{1}$,$\textcircled{2}$,$\textcircled{3}$), and

3. average performance over the test person combinations (P_1,P_2,P_3,P_4,P_5).

To evaluate the three offline baseline approaches with personal data P, we use a combination of LOPO and LODO scheme shown in Fig. 6.3 except we are not averaging over different training sequences. The reason is that for the offline approaches the order of training days is not considered. For the lower baseline of the fit(LOPO) approach, a LOPO scheme is used.

As the performance metrics for the online simulations, we use the **accuracy A** and **class-averaged F_1 measure** introduced in section 5.3, since the data set has a strong class imbalance shown in Fig. 2.4. As mentioned, the ratio between the two quantities discloses the minority and majority class performance. Due to the different classifier training and evaluation procedures, the offline and online results cannot be directly compared.

6.1.3 Online Learning Results

In the online simulation on the $\mathbb{D}_7$ set with seven subjects, we assess a possible A and F_1 performance improvement compared to the before introduced four baselines. This improvement is achieved by the daily model personalization updates of the initial person-independent LOPO model trained on six known subjects. The online training procedure is shown in Fig. 6.1 and works on the incoming either with the true user labels (true update) or own model predictions (pred update). The online simulation results are dependent on the training sequences. **One example of the online optimization** over the nine training days for one unseen test person is depicted in Fig. 6.4. The graph shows the mean hold-out performance of one fixed test day over the various daily training updates with true or predicted labels and a confidence interval of one standard deviation (std). Thus, this exemplary test subject has in total 10 recording days and one day is the test day in each cross-validation step. The final performance is then taken as average over all sequences after the last training day. Obviously, the training updates improve the recognition outcomes for true and predicted labels. These results depend on how similar the training days and the test day are so far, which also contributes to the seen std and it slightly grows during both updates. For training updates with the own classifier predictions, the results depend on whether the initial LOPO model is precise enough for the self-improvement. Otherwise, the model cannot improve or even degrade its performance, where no example is shown. This confirms the previous study of [65], which stated that the initial classifier model needs to have a certain level of performance to allow self-improvement during online learning.

Afterwards, the procedure is repeated for all combinations of test days and the performances are averaged. Then, the final **online F_1 and A results** are obtained for all classifiers in Table 6.1 and 6.2 by averaging over all subjects. For NB only the true label updates are able to improve the results over the initial LOPO model performance by F_1 and A rates of 0.5% and 0.3%. In contrast to the GMM, it slightly enhances both online updates in the accuracy measure and the F_1 rate stays constant except for the pred update with a small degradation of -0.1%. Thus, the GMM improves the majority class detection, whereas the SVM only recognizes the minority classes better. This is shown by the strong F_1 improvements of 2.4% and 1.1% for true and pred updates, but the accuracy measure decreases by -0.4% and -1.9%, i.e., the majority class detection is worse. The MLP classifier is not able

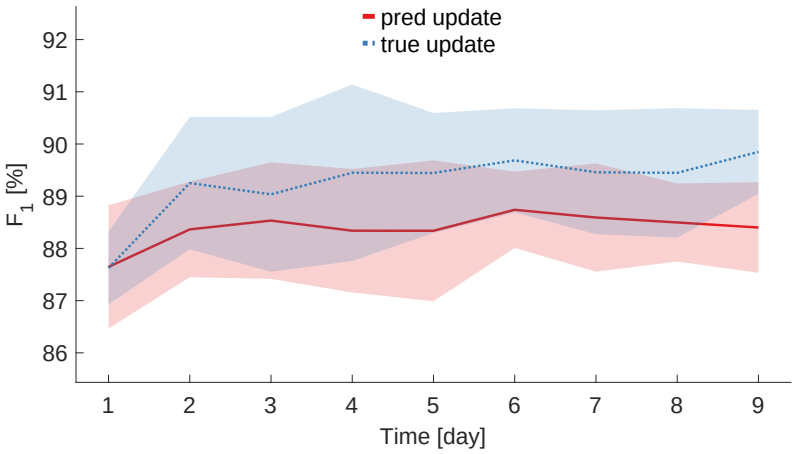


Figure 6.4: Mean hold-out performance of multiple training day sequences with a confidence interval of one standard deviation for the online RF classifier of one subject and test day.

Table 6.1: Results of online classifiers $\boldsymbol{F_1}$ performance [%], where the best baseline is in bold, and an improved online algorithm is cursive.

Training	Classifier	GMM	MLP	NB	RF	SVM
Upper Baselines	fit(LOPO+P)	71.0	**77.9**	65.1	**78.4**	69.6
	fit(LOPO)+adapt(P)	70.1	76.6	65.1	77.8	**71.6**
	fit(P)	**71.6**	75.4	**68.1**	76.8	71.5
Online	**true update**	70.0	75.8	*65.3*	*77.2*	*70.6*
	pred update	69.9	75.4	63.8	*76.5*	*69.3*
Lower Baseline	fit(LOPO)	70.0	76.3	64.8	75.6	68.2

Table 6.2: Results of online classifiers **A** performance [%], where the best baseline is in bold, and an improved online algorithm is cursive.

Training	Classifier	GMM	MLP	NB	RF	SVM
Upper Baselines	fit(LOPO+P)	78.1	**84.4**	73.4	**84.9**	80.1
	fit(LOPO)+adapt(P)	77.4	83.8	73.4	84.6	**81.5**
	fit(P)	**78.3**	82.6	**74.2**	83.7	81.0
Online	**true update**	*77.5*	82.5	*73.5*	*83.8*	78.6
	pred update	*77.4*	81.8	72.1	*83.2*	77.1
Lower Baseline	fit(LOPO)	77.3	83.2	73.2	82.8	79.0

Table 6.3: Standard deviation of different training day sequences averaged over all subjects (F_1 [%]).

Classifier	GMM	MLP	NB	RF	SVM
pred update	0.006	0.916	0.080	1.762	1.952
true update	0.005	2.564	0.000	1.520	3.796

to improve with both online updates. The best online and overall algorithm is the RF. It significantly improves its performance for both metrics and updates, i.e., the RF strongly enhances the F_1 and A rates up to 1.6% and 1%. Thus, the RF is the only classifier that profits from the online adaptation with a better minority and majority class detection. Therefore, the own model predictions are reliable enough that some classifiers can self-improve their performance.

The online adaptation process has some room for improvement in **comparison to the three personalized upper baselines**. For example, for the best online classifier, RF, the possible gain in F-measure and accuracy is 1.2% and 1.1% by comparing the true label update and the fit(LOPO+P) baseline. Therefore, these batch learning schemes with personal data perform better than the online updates, since the interday variations are high and different activities are carried out on several days. Thus, learning with more present classes and activities generalizes better over unseen data, which we partially addressed by adding the class representatives during the online training. Therefore, possible improvements can be made here in future investigations. The highest baseline for the GMM and NB are the personalized models (fit(P)) with solely the data of the person of interest. In contrast to the MLP, RF, and SVM classifiers, where the model improves more by having more data even from other subjects (fit(LOPO+P) and fit(LOPO)+adapt(P)). This training method generalizes better for these classifiers and marks the upper bound for the possible online improvement. For the NB, the order of fitting does not matter, since it updates only its count statistics and densities. That is why, fit(LOPO)+adapt(P) and fit(LOPO+P) have exactly the same performance of 65.1% for the F_1 score, but the true update is slightly improved by adding the class representatives during the online training.

Further analysis of the **training sequence impact** is depicted in Table 6.3, where the std of recognition rates over all training sequences is computed after the last update per person and is averaged over all subjects. Obviously, the GMM is stable with a very low std of 0.005% to 0.006%, since it only shifts the mean component. This update is independent of the order, because the vector sum is associative. The MLP has a 0.916% smaller std for its own predicted labels than true ones, because the model makes consistent predictions, but not necessarily right ones. This is also the case for the SVM. The linear hyperplane updates are more influenced with a std up to 3.796% by the order of training days, since the interday variability of different classes is also high. Thus, the hyperplane shifts are too sensitive with a small amount of data and do not generalize well. For the NB again, it updates only counts, where the order does not matter for the same result with the true labels. Thus, the standard deviation is very close to zero, but not exactly due to numerical reasons. But when using their own model predictions, the order slightly changes the models over time. Thus, the predictions produce the different count statistics, which explains the slightly higher std of 0.080%. The RF std is similar to the MLP, which comes from the used model construction. Since the RF has an inherent randomization by the data bootstrapping and random feature selection, the outputted trees are always different, which results in a slight std about 1.5%.

The systematic analysis of the **interday class prior variability** in Fig. 2.6 in section 2.3.2 shows that the distribution of daily class prior strongly changes across days. Thus, for example, the basics class has a highest range from 0% up to 70% with a medium value of 41.8%, i.e., on some days it does not occur at all,

but on other days it clearly is the majority of the routine activities. These high differences in the class priors across days create the previously mentioned challenges for the detection algorithms and hinder an easy adaptation to this non-stationary process. According to [61], these changes are the so-called concept drift, which can occur suddenly, reoccurring, or incremental. In our case, the most relevant changes happen on a recurrent basis, since a weekend with lots of free time activities strongly differs to a workday in office. A sudden change typically follows an event, e.g., an accident, which makes sport exercises impossible. That is why, the class distribution suddenly changes, whereas incremental drifts occur over a long period. For example, a new habit is established and every day the portion of sport exercises is slightly increased. Therefore, we do not observe any sudden nor incremental drifts.

To summarize the **findings on online learning** for daily routine recognition, it significantly improves the recognition performance of the random forest for the true and predicted label update. For other classifiers, the improvement effect is less strong, since the initial fitted start model is not as precise as for the RF. Thus, the online learning is, in particular, beneficial for the RF classifier with both updates and the person-independent start model. Obviously, the enhancement is stronger with the true user feedback than the self-improvement with the own model predictions. A recommended application would be a combination of both updates by incorporating the certainty of prediction, namely the posterior probability, to only ask for a user feedback if the classifier is uncertain [139].

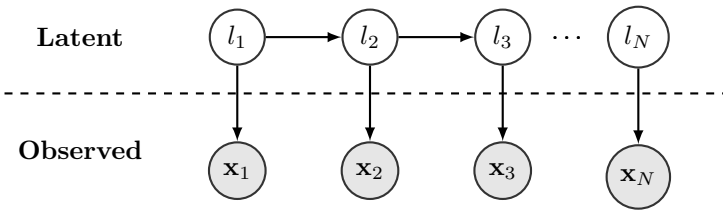


Figure 6.5: The HMM classifier decodes the latent state sequence generating the observed data.

6.2 Modeling the Temporal Daily Routine Transitions by Sequence Learning

To model the temporal relationships of neighboring samples, we introduce various sequence learning techniques in section 6.2.1. These approaches can exploit the recurring temporal sequence of the daily routine situations and environments. The routine activities have a temporal correlation in their feature space as well as in the respective routine class. These patterns repeat over time. That is why, the expected benefit of sequence classifiers is a higher temporal stability of the predictions in comparison to the online learning, which should result in an improved classification. The evaluation and experimental setup are explained in section 6.2.2, which determines the sequential classification results in section 6.2.3.

6.2.1 Sequence Learning Techniques

With the found feature vector $\mathbf{x}$ of section 5.1, we classify the routine behavior and environments by exploiting the sequence characteristics. Consequently, we take advantage of the time correlation of feature vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ and labels $l_1, l_2, \ldots, l_N$ during the learning process, where a label l is chosen out of the K classes $\mathbb{C} = \{c_1, \ldots, c_K\}$ for all of the N samples. For this task, the sequence learners, hidden Markov model (HMM) with different observation models and long short-term memory (LSTM) network, are selected for the evaluation, which are computationally feasible to use in a HA. We perform sequence learning on the entire training data and apply the fixed model on the unknown test data for the evaluation.

HMM describes the observed temporal sequences of feature vectors as outcomes of hidden states generating these observations as shown in Fig. 6.5 [69]. Since in our supervised case the hidden states correspond to the classes, the HMM is represented by the joint probability distribution [140]:

$$p(l_1, \ldots, l_N, \mathbf{x}_1, \ldots, \mathbf{x}_N) = p(l_1) \prod_{n=2}^{N} p(l_n|l_{n-1}) \left[\prod_{n=1}^{N} p(\mathbf{x}_n|l_n) \right]. \tag{6.9}$$

This simplifies the learning procedure of the three quantities on the training data:

- The **initial probability distribution** $p(l_1)$ describes the probability to start in a certain class. This is set to the uniform distribution with probability $\frac{1}{K}$

that the observation model solely determines the class decision for the first sample, i.e., $p(l_1, \mathbf{x}_1) = \frac{p(\mathbf{x}_1|l_1)}{K}$. Alternatively, $p(l_1)$ can be determined by the class prior $p(l)$, which is estimated by the frequency of class labels.

- The **transition probability** $a_{ij} = p(l_n = c_j|l_{n-1} = c_i)$ defines the probability to switch from class c_i to c_j and is estimated by the maximum likelihood (ML) approach. That means the expected number of transitions from c_i to c_j is divided by the expected number of times c_i occurs. Two example transition graphs are shown in Fig. 6.12 and 6.16.

- The **observation probability** $p(\mathbf{x}_n|l_n)$ expresses the class likelihood that a feature vector is generated by a class.

We use three different models to generate the observation probabilities, which are the following classifiers trained in a supervised learning scheme and the same parameters explained in section 5.2:

- random forest ensemble of 20 decision trees,

- multi-layer perceptron with a hidden layer consisting of 100 neurons, and

- Gaussian mixture model with a mixture model of 8 components per class and a diagonal covariance matrix.

These classifiers decide for the routine class that has the maximal posteriori probability $p(l_n|\mathbf{x}_n)$. Hence, we compare the recognition performance of the sole classifier model against the combination with an HMM. To apply these classifiers as observation models in Equation (6.9), we need to convert their output to the class likelihood

$$p(\mathbf{x}_n|l_n) = \frac{p(l_n|\mathbf{x}_n)p(\mathbf{x}_n)}{p(l_n)} \tag{6.10}$$

via the Bayes rule [141]. The evidence term $p(\mathbf{x}_n)$ is a constant and can be ignored in the decoding of the most likely class sequence, which is given by

$$\arg\max_{l_1,\ldots,l_N} p(l_1, \ldots, l_N, \mathbf{x}_1, \ldots, \mathbf{x}_N). \tag{6.11}$$

According to [69], the Viterbi algorithm decodes the most likely sequence by setting (6.10) in (6.9), taking the logarithm, and ignoring constant factors:

$$\begin{aligned} &\log\, p(l_1, \ldots, l_N, \mathbf{x}_1, \ldots, \mathbf{x}_N) \propto \\ &\sum_{n=2}^{N} \log\, p(l_n|l_{n-1}) + \left[\sum_{n=1}^{N} \log\, \frac{p(l_n|\mathbf{x}_n)}{p(l_n)}\right], \end{aligned} \tag{6.12}$$

which can be rewritten in a recursive way:

$$\begin{aligned} \omega(l_{n+1}) = &\log\, \frac{p(l_{n+1}|\mathbf{x}_{n+1})}{p(l_{n+1})} + \max_{l_n} \left[\log\, p(l_{n+1}|l_n) + \omega(l_n)\right] \\ &\text{with initialization } \omega(l_1) = \log\, \frac{p(l_1|\mathbf{x}_1)}{p(l_1)}. \end{aligned} \tag{6.13}$$

This allows to find the most likely sequence for each time step and the optimal sequence is found by backtracking the gone steps. For an optimal performance, we stored the sequence of one day and did the backtracking on this sequence. In a future online application, storing of longer sequences is not possible due the high memory load. That is why, an online variant or a short-term Viterbi decoder need to be used [142, 143, 144].

LSTM is capable of learning long-term relationships in data sequences [71, 72]. To do so, the LSTM can memorize information over sequential time steps by introducing a cell state vector $\mathbf{c}_n$ and hidden state vector $\mathbf{h}_n$ as its output [145]. The information can flow through the network as shown in Fig. 6.7 by the cell states, which can be updated by the forget gate output $\mathbf{f}_n$, cell candidate output $\mathbf{g}_n$, and input gate output $\mathbf{i}_n$:

$$\mathbf{c}_n = \mathbf{f}_n \odot \mathbf{c}_{n-1} + \mathbf{g}_n \odot \mathbf{i}_n, \tag{6.14}$$

where $\odot$ denotes the Hadamard product (element-wise multiplication). The forget gate controls how much information of the previous cell state is kept and forgotten, whereas the cell candidate and input gate include new information of the current sample $\mathbf{x}_n$ to update the cell state. The cell state vector allows with the output gate output $\mathbf{o}_n$ to define the output of the LSTM $\mathbf{h}_n$:

$$\mathbf{h}_n = \mathbf{o}_n \odot \tanh(\mathbf{c}_n). \tag{6.15}$$

Therefore, the LSTM has three separate processes – forget, update, and output – to control the information flow and memorize the temporal relationships in the data sequences. For each gate and the cell candidate, the input and hidden state vector $\mathbf{x}_n$ and $\mathbf{h}_{n-1}$ are multiplied by the weights and a bias term is added. Afterwards, a sigmoid activation function σ_g is used for the gates and tanh activation is applied for the cell candidate:

$$\mathbf{f}_n = \sigma_g(\mathbf{W}_f \mathbf{x}_n + \mathbf{W}_f^R \mathbf{h}_{n-1} + \mathbf{b}_f), \tag{6.16}$$

$$\mathbf{i}_n = \sigma_g(\mathbf{W}_i \mathbf{x}_n + \mathbf{W}_i^R \mathbf{h}_{n-1} + \mathbf{b}_i), \tag{6.17}$$

$$\mathbf{o}_n = \sigma_g(\mathbf{W}_o \mathbf{x}_n + \mathbf{W}_o^R \mathbf{h}_{n-1} + \mathbf{b}_o), \tag{6.18}$$

$$\mathbf{g}_n = tanh(\mathbf{W}_g \mathbf{x}_n + \mathbf{W}_g^R \mathbf{h}_{n-1} + \mathbf{b}_g), \tag{6.19}$$

where the trainable parameters are the biases $\mathbf{b}_\cdot$, weights $\mathbf{W}_\cdot$ and $\mathbf{W}_\cdot^R$ for the gates and the cell candidate.

We test these long-term memory capabilities in the routine domain, where activities last for longer periods of minutes to hours. To evaluate these recurrent networks, we need to specify the network architecture and further hyper-parameters such as the number of neurons per layer. Thus, we analyzed the effects of a fully-connected (FC) pre-layer and post-layer around the LSTM layer with a softmax output unit, which is illustrated in Fig. 6.6. Hereby, we varied the number of neurons for all layers in a range from 32 to 128 units. After an empirical architecture evaluation with a L2-regularization to decay the weights, an Adam optimizer, and an early-stopping criterion, the LSTM layer started very early to overfit and learned the training data by heart if it contained too many neurons. Thus, we optimized the final architecture and the number of neurons to learn the sequence relationship while keeping the network as small as possible to avoid overfitting. The results show that the net with

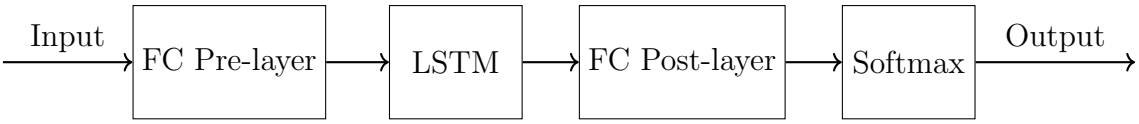


Figure 6.6: The LSTM network consists of a fully-connected (FC) pre-layer and post-layer around the LSTM unit plus a softmax output stage.

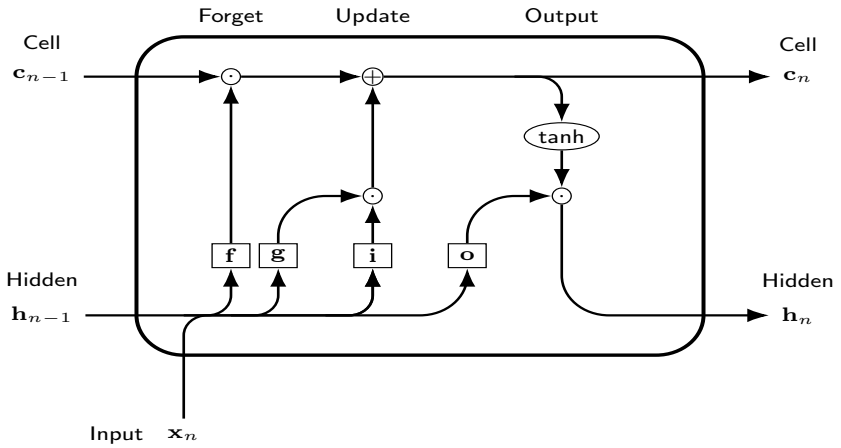


Figure 6.7: The LSTM unit is shown for one-time step, where the overlapping lines of input $\mathbf{x}_n$ and hidden vector $\mathbf{h}_{n-1}$ stand for a concatenation $[\mathbf{x}_n; \mathbf{h}_{n-1}]$. The unit has three gates: forget $\mathbf{f}$, input $\mathbf{i}$, and output $\mathbf{o}$ plus the cell candidate $\mathbf{g}$ to control the information flow through it and update on new input data.

64, 64, and 32 neurons for pre-layer, LSTM-layer, and post-layer is the best and we use this network for our evaluation.

We performed the experiments in MATLAB R2019b and used the LSTM (from the deep learning toolbox), self-implemented HMM and GMM classifiers with the fitting functionality of Python library scikit-learn 0.22.2 for the MLP network, RF trees, and GMM probability distribution [134].

6.2.2 Sequence Learning Evaluation and Experimental Setup

For the supervised sequence evaluation, we assess the model performance based on a **cross-validation** (CV) scheme. Since the Huynh data set only contains one person, we can only perform a leave-one-day-out scheme, i.e., the personalized model capabilities are evaluated on the seven weekdays. In contrast, in the $\mathbb{D}_7$ data set we can assess the model generalization abilities across the seven subjects with a leave-one-person-out scheme. In general, a personalized model has a better performance than a person-independent one [44], which we also showed for the high-level daily routine recognition in section 5.4 [10]. We are interested in the model generalization abilities of sequence models across multiple subjects. To deal with the strong class-imbalance in both data sets, we require recognition metrics that make it obvious if the classifier ignores the minority classes [146]. Thus, we compute the **confusion matrix**, the **accuracy** (A), and the **class-averaged F-measure** F_1 introduced in section 5.3.

To further consider the temporal order of the predictions and ground truth, we apply the segment error assignments of [147]. It defines a segment as a change in

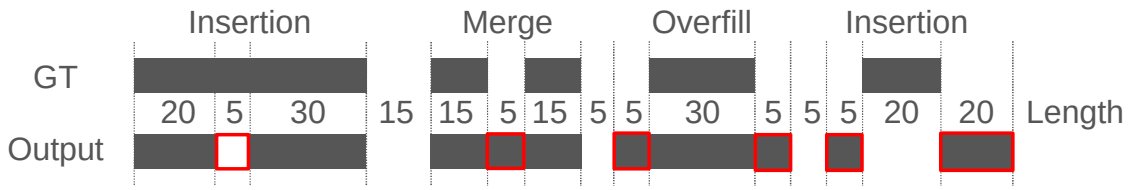


Figure 6.8: The segment error types are shown for a binary classification example with the ground truth (GT) and an output of a classifier. The vertical dashed lines denote the segment boundaries, and the red rectangles mark the erroneous segments. For the exemplar segment error calculation in Table 6.4, the length of each segment is given in the number of samples.

the ground truth or prediction output. Thus, the three error types are computed as illustrated in Fig. 6.8:

- **Insertion** is a wrong class transition at the start or end of a segment or fragments a segment of the same class in parts. For example, before or after a social class segment, the classifier wrongly predicts a physical segment instead of transportation or within a social segment the predictions change to the listening class. Thus, a high number of insertions means a classifier is not stable in time and often changes its predictions between the classes.

- **Overfill** is a segment that extends a ground truth segment over its boundaries, i.e., a class segment starts too early or ends too late. That is why, a high number of overfills stand for a classifier that changes its predictions too less.

- **Merge** is a special case of an overfill, where between two occurrences of the same class no change to another class happens. For example, the ground truth has a sequence of social, listening, and social, but the classifier just outputs social.

Since the segments in both data sets have a variable length, we normalize the three error types per sample duration and not per number of segments, which would be misleading, since the variable segment lengths have a different weight. Since we face a multi-class problem, the output is a special kind of confusion matrix, the so-called segment error table. That is why, we simplify the analysis by summing up the error patterns over all classes. Therefore, we do not distinguish if, e.g., an insertion error happened for one class and not the other. Thus, the sum of these three segment errors is 1 minus the accuracy value. Based on the three segment errors, we can judge a classifier's tendency to change the predictions over time a lot or be stable. An exemplar segment error calculation is explained in Table 6.4 based on the classifier output in Fig. 6.8. It shows as mentioned the misleading effect of a variable segment length on the output of the segment error calculation normalized by the number of segments. For example, only 5 samples have a merge error representing only 2.5% of the total sample duration, but the proportion normalized by segments would be 6.25%. That is why, we use the sample duration for the normalization to avoid the over-weighting of short segments.

Table 6.4: Exemplar calculation of the segment errors of Fig. 6.8.

Type	Sample duration	Percentage	# of Segments	Percentage
Insertion	30	15%	3	18.75%
Merge	5	2.5%	1	6.25%
Overfill	10	5%	2	12.5%
Correct	155	77.5%	10	62.5%
Sum	200	100%	16	100%

6.2.3 Sequence Learning Results

The DRR results for the LSTM and HMM sequence learners with three observation models are compared against the performance of classifiers without exploiting the temporal relationship on two data sets, Huynh and our set $\mathbb{D}_7$.

Huynh Set

Starting the leave-one-day-out CV analysis on the Huynh set with acceleration data and the time of day feature, we show the personalized classifier evaluation with recognition rates and segment error analysis plus the confusion matrix of the best performing algorithm in Fig. 6.9, 6.10, and 6.11. We first analyze the **performance of the non-sequence classifiers used as the HMM observation models** in Fig. 6.9 without modeling the sequence relationships, and then check for a possible improvement marked as the red bar by sequence learning approaches. It is obvious that the RF classifier is the best non-sequence learning model with an F_1 and accuracy performance of 86.6% and 93.0% compared to GMM and MLP with an accuracy of 87.8% and 87.4%. However, they both have detection problems with the minority classes resulting in a lower F_1 rate of 71.8% and 75.4%, since the minority classes strongly overlap within the ACC space. The MLP has a higher capability to model a complex decision boundary than the GMM, which explains the better minority class detection. In this case, further features such as audio could ease the detection problem. The reason for the big performance gap between the classifiers is, that the decision trees of the RF can effectively profit from the time of day feature. The decision trees can derive rules like from 12 a.m. to 1 p.m. it is lunch, because the Huynh set has a very structured daily routine.

After **adding the HMM sequence learner** to the three classifiers, all metrics improve, which is shown by the red bar in Fig. 6.9. The GMM-HMM classifier has the smallest F_1 and A increase of 1.2% and 1.1%, because it has the most problems with the class overlap in the feature space. Then, the RF-HMM combination follows and strongly improves by 5% and 2.8%. The MLP-HMM classifier nearly doubles both metrics by 10.2% and 5%. However, the MLP model starts from a lower level of correct decisions than the RF. Thus, adding the HMM gives more possibilities to smooth out erroneous class transitions. In an overall comparison, the RF-HMM combination is the best classifier even compared to the LSTM, which has an F_1 and A margin of 13.9% and 9.4% compared to the best. To enhance the LSTM, the amount of training data needs to be increased for a better model generalization. The ranking stays the same as without the sequence modeling since the classes overlap too much within acceleration feature space and RF is the only classifier that can

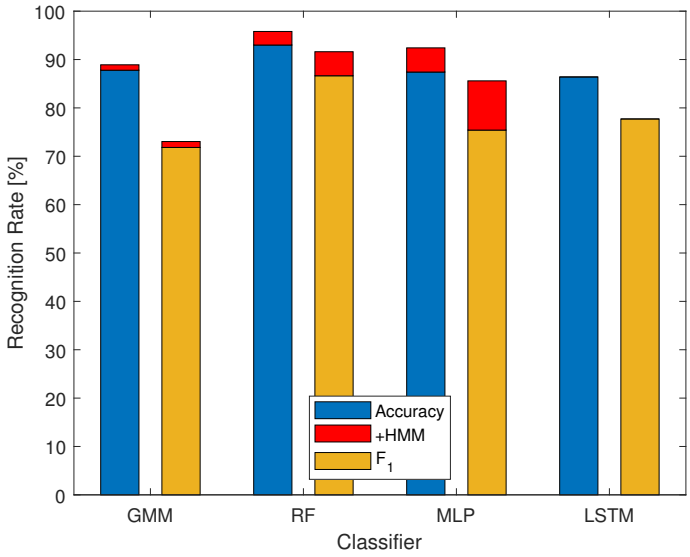


Figure 6.9: Classifier performance evaluation on Huynh set, where the red bar denotes the improvement of an algorithm by adding the sequence learner HMM.

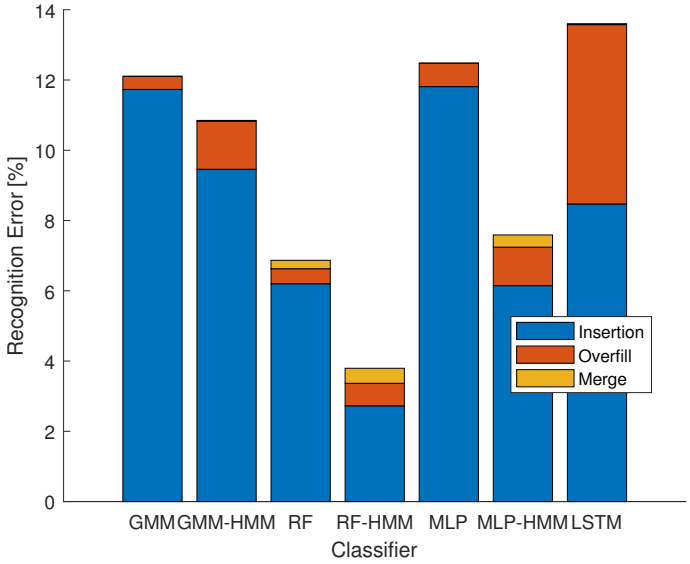


Figure 6.10: Classifier segment error evaluation on Huynh set.

effectively exploit the time of day feature.

In comparison to the **prior work of Huynh** [18], we outperform both of his methods, GMM-HMM and TM, with an F_1 rate of 64.6% and 74.3%, i.e., our GMM-HMM and RF-HMM have an F_1 rate of 73.0% and 85.6%. The high margin is a result of our well-performing high-level feature representation and the appropriate window length of 1 minute, where Huynh used a length of 30 minutes.

Analyzing the **time behavior of predictions**, the results of the segment evaluation are depicted in Fig. 6.10. The main source of errors are insertions with the highest number of cases for the non-sequence classifiers, GMM and MLP, with 11.7% and 11.8%. Adding the HMM highly decreases the insertion percentage by 2.2% and 5.7% while slightly increasing the overfill error by 1% and 0.4%. Thus, the stability of classifier predictions is improved as expected, which increases the number of overfill events. This is also the case for the RF, which has a similarly

Figure 6.11: Confusion matrix of the best-performing RF-HMM sequence learner on Huynh set.

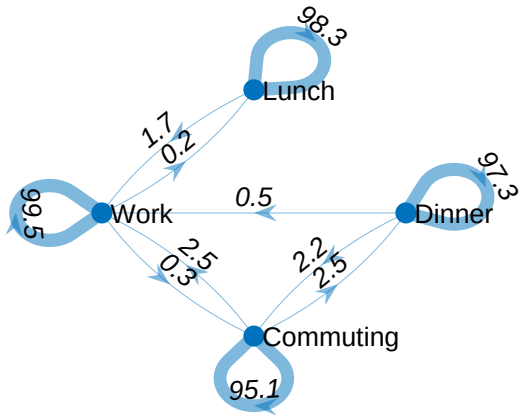


Figure 6.12: Transition graph of first CV fold on Huynh set [%].

low quantity of insertions like the MLP-HMM and even lower for the RF-HMM with the best result of 9.5%. The reason is that the decision trees of the RF can efficiently deal with the time of day feature, which has a very high predictive power for the Huynh set due to the structured daily work routine. Merge errors only occur for the RF, RF-HMM, and MLP-RF with a small percentage of 2.7% to 6.2%, because of the mostly long class duration, which makes it difficult to merge segments. The LSTM has the worst performance with the biggest overfill error of 5.1% and a medium insertion error of 8.5%, since it has a strong tendency to stay with its class predictions over a longer period and changes them too less. Therefore, the sequence learning approaches improve as expected the temporal stability of the predictions. In some cases, the temporal stability is too strong for the LSTM algorithm.

Furthermore, we analyze in detail the **confusion matrix** of the best-performing RF-HMM, where the class-wise recall is shown in the rows in Fig. 6.11. The majority class, "work", is particularly well detected with a high recall of 98.7%. Only a few errors occur due to other situations, which also consist of seated activities, e.g., lunch or dinner, since the activity patterns of the ACC data are very similar. Here,

different sensors could be beneficial to distinguish these kinds of situations. Some confusions, such as between lunch and commuting or dinner and lunch, do not happen, even though they could have similar activity patterns. This is the case, since Huynh's working routine is very structured and the classes contain an implicit time order: commute, work, lunch, work, commute to dinner. This knowledge is also present in the transition matrix of the HMM, since some transitions, e.g., lunch to dinner, do not happen. To show this switching behavior between the classes, we plot the **transition graph** of the first cross-validation fold in Fig. 6.12, where the weight of a transition parameter a_{ij} is displayed as the thickness of an arrow. No transition event between two classes corresponds to no arrow in the graph, e.g., between lunch and commuting. Obviously, the strongest transition is staying in the same class, e.g., 97.3% for dinner, i.e., the probability mass is strongly concentrated on the diagonal elements of the transition matrix due to the long duration of routine events. Thus, the duration density for each class is uniformly distributed with a small spread. Therefore, it is not optimal for an HMM, which models the transitions to decay exponentially [69]. The only exception occurs for commuting that has a peak at the typical period.

Our Data Set

On the contrary to the Huynh set, we perform a leave-one-person-out CV and test the person-independent model generalization across subjects on the acceleration and audio data. The results of the classifier evaluation are depicted in Fig. 6.13 and 6.14 as well as the confusion matrix of the best performing algorithm in Fig. 6.15.

Again, we start the analysis with **the performance of the non-sequence classifiers** in Fig. 6.13 that are used for the observation models. Afterwards, the possible gain of sequence modeling is assessed. The RF classifier is the best non-sequence learning model with an F_1 and accuracy performance of 79.4% and 83.9%. These rates are slightly superior to MLP with 78.6% and 83.2%. The GMM lies within a margin of 6 to 7% in both metrics. In comparison to the Huynh results, we see a lower overall performance due to the person-independent training and the more complex problem. The minority class recognition works relatively better because of the smaller difference between the accuracy and F_1 metrics. Here, the rich audio features are beneficial to distinguish the routine classes.

To assess the **gain of sequence modeling**, we use the three classifiers as observation models for the HMM and all metrics strongly improve about 4 to 7%. Thus, the best non-sequence classifier, RF, enhances the rates about 4% by including the HMM, whereas the GMM-HMM gains an upgrade of almost 5%. The MLP-HMM has the strongest F_1 and A improvement of 6.7% and 5.4%. Thus, the MLP-HMM combination is the winner even against the LSTM, which has an F_1 and A margin of 3.8% and 3.5% compared to the best. To enhance the LSTM performance, the amount of training data needs to be increased for a better model generalization. Additionally, the sequence training in mini-batches could be further optimized. According to [72], training on long sequences can be problematic, since a sufficiently large LSTM may memorize the entire sequence resulting in a bad model generalization. We analyze the long-term memory effect by visualizing the output scores of the LSTM and notice a stronger tendency to stay with its class predictions over a longer time interval.

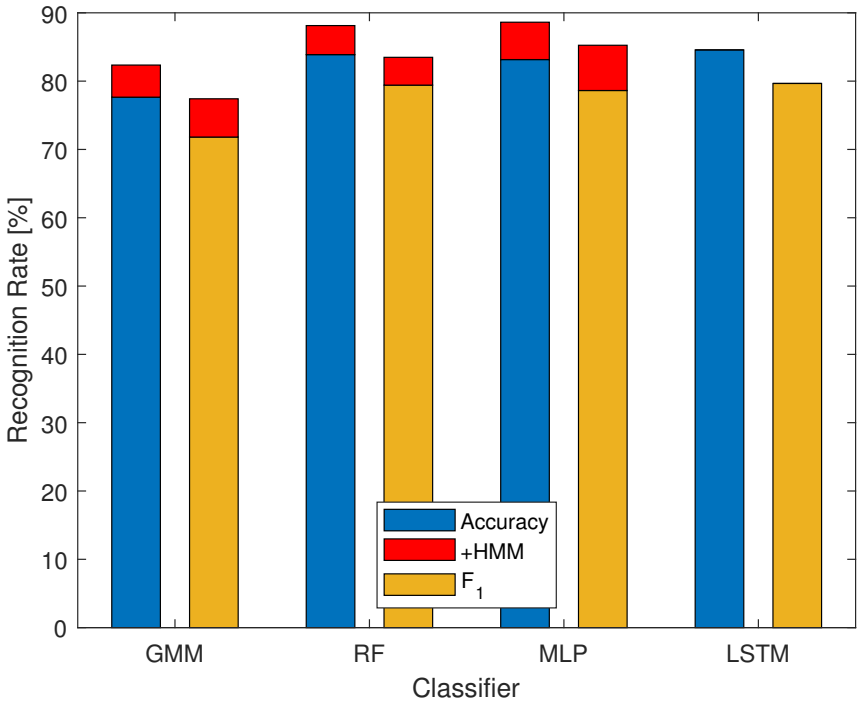


Figure 6.13: Classifier evaluation on data set $\mathbb{D}_7$, where the red bar denotes the improvement of an algorithm by adding the sequence learner HMM.

Analyzing the **time behavior of predictions**, the results of the segment evaluation are depicted in Fig. 6.14. The main source of errors are insertions with the highest number of cases for the non-sequence GMM classifier with 20.9%. Adding the HMM to the GMM, it highly decreases the insertion percentage by 6.8% while slightly increasing the overfill error by 1.7%. Thus, the GMM-HMM has a similar level of insertion errors like the RF and MLP of about 14-15%. The RF and MLP including the HMM strongly decrease the insertions by 9.1% and 9.5% while enhancing the number of overfills by about 4%. Thus, the stability of classifier predictions is improved as expected, which increases the number of overfill events. The best overall performance is achieved by the MLP-HMM. Merge errors only occur for the RF and the sequence learners with a small percentage of 0.2% to 2.2%. The LSTM has a medium overfill performance of 4.2% and a medium insertion error of 10.5%, since it has a strong tendency to stay with its class predictions over a longer period and rarely changes.

Furthermore, we analyze in detail the **confusion matrix** of the best-performing MLP-HMM in Fig. 6.15. Obviously, the two majority classes, social and basics, and transportation are very well recognized with a recall over 90% and contribute to the high overall accuracy of 88.1%. They are mainly distinguishable through audio characteristics, such as low-frequency car noise or own voice activation. In contrast, the strong confusion between listening and basics (16.4%) or social (14.8%) stems from the high similarity within the audio features and the strong dependency of the reference class on the subjective user intention. This means a background conversation can be either listening or basic depending on if the subject wants to follow it. Additionally, it can quickly change to social if the subject decides to participate in the conversation. Thus, we have many transitions between these classes and, in general, all routine transitions are possible, which is different to

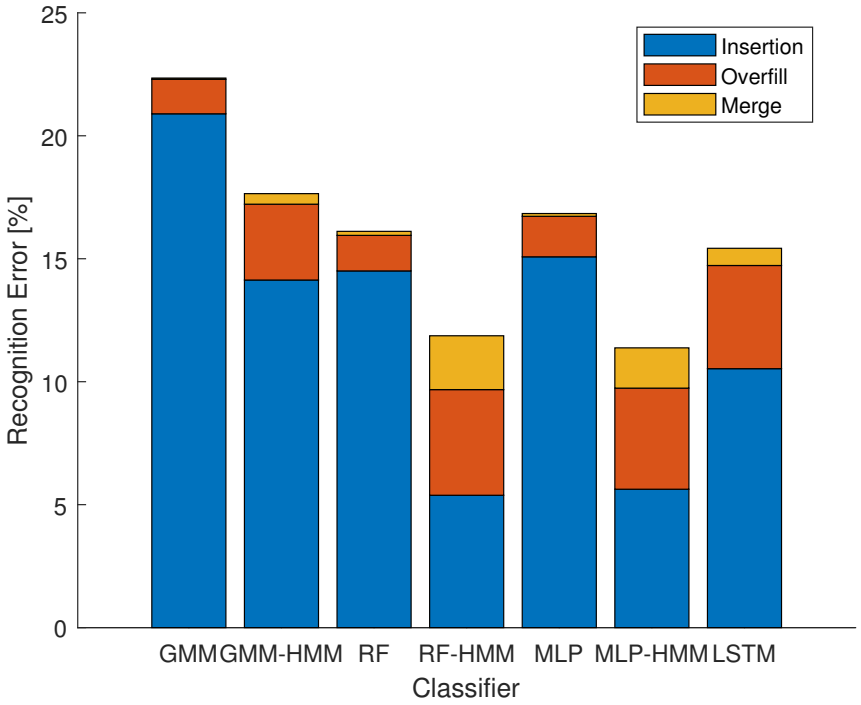


Figure 6.14: Classifier segment evaluation on data set $\mathbb{D}_7$.

the Huynh set. To show this switching behavior between the classes, we plot the **transition graph** of the first cross-validation fold in Fig. 6.16. Obviously, the strongest transition is staying in the same class, e.g., for social 96.7%, i.e., the main portion of the probability mass lies on the diagonal elements of the transition matrix. However, on some days, not all transitions happen since an activity class is not performed every day. The HMM inherently models a duration probability density that is exponentially decaying [69], which is a good fit to our data set. This is, because many events have a short duration of a few minutes, and a small number of events have a long duration of hours. Similarly, during sport activities, we have a high intensity in the ACC signal and a voice activation, which leads to the bigger mismatch of 17.3% between physical and social. Thus, both classes can also occur simultaneously and then the user's intention decides. Here, the situational intention needs to be better decoded from suitable motion patterns or further sensors, e.g., electromyograms for listening attention [148], that can deliver a more reliable input.

To summarize the **findings on sequence learning** for daily routine recognition, it strongly improves the recognition performance of all tested non-sequence learners, RF, MLP, and GMM by adding the HMM to them. For the GMM classifier, the improvement effect is less strong than for the others. The RF and MLP outperform the LSTM model. Thus, the sequence learning is, in particular, beneficial for the RF and MLP classifier as the observation model for the HMM. In addition, we noticed the expected improvement in the prediction stability over time. A recommended application would be to apply either the RF-HMM or MLP-HMM combination.

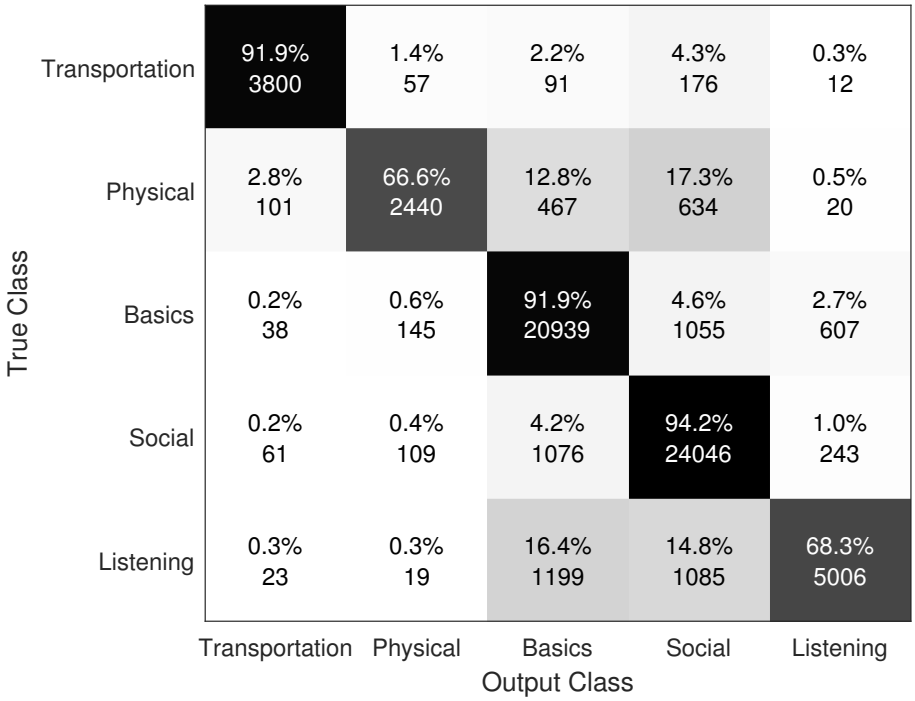


Figure 6.15: Confusion matrix of the best-performing MLP+HMM sequence learner on data set $\mathbb{D}_7$.

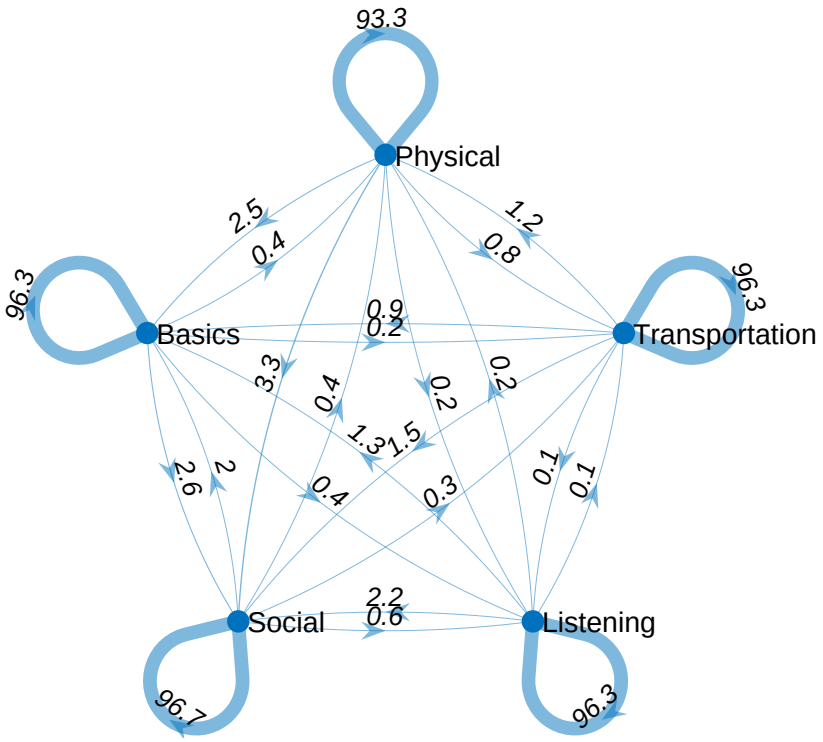


Figure 6.16: Transition graph of the first CV fold on set $\mathbb{D}_7$ [%].

6.3 Improved Classification Summary

In this chapter, we improved the daily routine recognition (DRR) on two real-world data sets $\mathbb{D}_7$ and $\mathbb{D}_H$ using online and sequence learning methods. The online approaches adapted the classification models over time based on the incoming data stream, whereas the sequence classifiers modeled the temporal relationships between neighboring samples. On this basis, we performed a comprehensive online and sequence classifier evaluation to compare the model performances. A possible combination of online and sequence learning was not analyzed, but it stays open for future investigations, and it is expected to be beneficial. Thereby, the RF classifier outperformed the other methods in online learning and also strongly performed in sequence learning. Thus, the RF is the recommended approach and is expected to profit the most from a combination of online and sequence learning.

Summary

- In our online simulations, the random forest (RF) enhanced the F- and accuracy rates up to 1.6% and 1% using the true and predicted labels compared to the baseline of the initially fitted model.
- Thus, the RF classifier was able to self-improve its model over time and the improvement was significant. Other classifiers only enhanced either the minority or majority class detection.
- Additionally, we analyzed the effect of the training sequence order and demonstrated a smaller influence of less than 2% at the F-measure rate for the RF.
- In sequence learning experiments, we demonstrated that the multilayer perceptron (MLP) and random forest observation model for hidden Markov model (HMM) achieved the best F-measure performance of 85.3% and 91.6% on our set and the Huynh set.
- Thereby, the MLP has the strongest F-measure improvement of 6.7% and 10.2% on both sets by adding the HMM. The long short-term memory network has an F-measure of 79.7% and 77.7% on both sets.
- The segment error analysis discovers for sequence learners the improved prediction stability over time. On our set, the remaining confusion for classifiers mainly stems from the intention-based class decision of the HA users.

Chapter 7

Conclusions and Outlook

In this dissertation, we are first to propose a personalized classification system for hearing aids (HA) based on daily routine recognition (DRR) on microphone and accelerometer data. Thereby, the goal was to improve a state-of-the-art person-independent HA classification system by ensuring the temporal stability of the predictions plus considering the user behavior and acoustic situations. While HA classification systems are typically trained on recordings of selected acoustic situations in real conditions and controlled laboratory environments with only microphone data, we solely performed our analysis on realistic unconstrained situations and environments of HA wearers following their personal daily routine. Thus, our achievements grouped into the creation of two real-world data sets, the clustering and visualization analysis of routine situations, the feature impact evaluation of microphone and acceleration (ACC) data, labeling the daily routine data with an extended semi-supervised approach, the offline, online, and sequence routine classification investigations.

In the following, we summarize the corresponding achievements in this dissertation in section 7.1 and give an outlook for the future work in section 7.2, where the explained material is partially published in [9, 10, 11].

7.1 Conclusions

While developing and optimizing a personalized daily routine recognition system several achievements of this dissertation can be mentioned:

First of all, to perform the routine analysis, a data set is needed, and the public Huynh set $\mathbb{D}_{\mathrm{H}}$ with two accelerometers on the wrist and in the pocket exists. It contains the real life of Huynh over seven working days. Since we are dealing with hearing aids, the sensor location is fixed to the ear position on the head, which is not the case for the Huynh or other activity sets. In addition, the existing previous studies did not use rich audio features over long-term periods in this high sampling rate as we do. Hence, there was the need to construct the realistic data sets, $\mathbb{D}_{\mathrm{T}}$ with one person and $\mathbb{D}_7$ with multiple subjects. Therefore, one of our main contributions is the time-consuming **creation of two large real world data sets** with acoustic and acceleration data. The set $\mathbb{D}_{\mathrm{T}}$ is recorded for the feasibility experiments to show which situations are distinguishable and analyze the impact of ACC and audio features. It has a length of 4016 minutes over 9 days with one subject. To compare the results with the prior work on the public $\mathbb{D}_{\mathrm{H}}$ set, we used similar generic

activity labels of a coarse time diary. With the help of data set $\mathbb{D}_7$, we assessed the model generalization abilities for high-level activities across the seven subjects. Since the very young study participants are non-representative for HA wearers, they are expected to have a more active lifestyle, which makes our task more challenging. Thereby, we analyzed the detection performances in a personalized and person-independent training while using audiological relevant intention-based hearing routine annotations given by HA users. The set contains over 104 days or 63449 minutes of recorded data. Both large data sets are recorded in unconstrained environments and have various real-life activities of subjects following their personal daily routine. Thereby, we bridged as one of our main contributions the literature gap to perform several further investigations introduced in the following.

In our clustering and visualization experiments on the data sets $\mathbb{D}_{\mathrm{T}}$ and $\mathbb{D}_{\mathrm{H}}$, we analyzed which situations are distinguishable within the feature space. We demonstrated on the $\mathbb{D}_{\mathrm{T}}$ set that **visualization** plots of ACC and audio features or the activity-loudness map already show **distinguishable routine behavior over time**. Additionally, applying dimensionality reduction techniques on the statistical feature representation for visualization purposes on both sets, the evaluation demonstrated that the t-distributed stochastic neighbor embedding (**t-SNE**) finds a very **meaningful embedding** of the high-dimensional data. Using the hierarchical clustering method on the t-SNE projection of the $\mathbb{D}_{\mathrm{T}}$ data, the found clusters, working, listening, and talking, form a continuous manifold with similar labels confirming the **manifold assumption**. In addition, the verifying the **clustering assumption**, that, e.g., sport or transport group form own clusters in the feature space. Therefore, we demonstrated that various situations are distinguishable by the ACC and audio feature representation.

Labeling the daily routine data of the set $\mathbb{D}_{\mathrm{T}}$, a semi-supervised algorithm propagated the coarse knowledge of a time diary to remaining larger set of samples. One of our contributions demonstrated that our extension of the visual interactive labeling (VIL) method works well for **consistent data annotations** in context of daily routine recognition. It offers the advantages, such as spotting of short routine events or a better handling of time-offsets with a coarse time diary. Additionally, the extended VIL approach is highly capable for **data exploration** purposes. The VIL method is validated on the public Huynh data set with only acceleration data and showed a weaker performance compared to the $\mathbb{D}_{\mathrm{T}}$ set, because the routine classes overlap within the feature distributions. For the optimal VIL performance, the classes need to be differentiable in the feature space, which we could show in the clustering analysis for the $\mathbb{D}_{\mathrm{T}}$ set.

To find an optimal feature representation for the DRR on the set $\mathbb{D}_{\mathrm{T}}$, we analyzed the impact of our features for the classification tasks. That is why, the DRR is performed on the three input sets: acceleration, audio, and ACC plus audio. With only the ACC data, only for some routines a good result is achieved, but others need to be improved by a better representation. One of our contribution enriches situational details and confirms that our **audio features** are **very informative** and improve the performance of routine classification in comparison to only applying ACC features. In addition, we demonstrated that our selected efficient feature representation is beneficial to **differentiate various daily routine situations**

and environments. Furthermore, we showed on the set $\mathbb{D}_7$ that our statistical feature representation is robust against the missing feature problem, since due to the Bluetooth data transmission, the statistical features are affected by a varying number of samples.

For the offline daily routine recognition on high-level activities, the model generalization investigations are performed on our data set $\mathbb{D}_7$ of seven people featuring ACC and additional audio data with different cross-validation schemes. We confirmed that the **personalized model is superior** to person-independent classifier. We demonstrated for high-level activities that the leave-one-fold-out cross-validation returns over-optimistic results due to the temporal correlation bias of neighboring samples in different folds. We further showed that the **best classifiers, multi-layer perceptron (MLP) and random forest (RF),** yielded the significantly best F-measure performance. The remaining misclassified samples require a tailored motion representation to distinguish the intended behavior more precisely. In addition, we also processed the Huynh data set with our supervised scheme and outperformed the topic model approach.

In our online simulation on our data set $\mathbb{D}_7$, the goal was to improve the DRR performance. A classifier was initially trained on the known subjects and then fine-tuned on the incoming data of the test subject. The online adaptation is performed to personalize the model with the true user or predicted labels. Thereby, the random forest strongly enhanced the F- and accuracy rates using the true and predicted labels compared to the baseline of the initially fitted model. Thus, the **RF classifier** can significantly **self-improve** its model over time. Other classifiers only enhanced either the minority or majority class detection and had a weaker performance. Additionally, we analyzed the effect of the training sequence order and demonstrated a smaller influence at the F-measure rate for the RF. The stronger interday variations of the active young subjects hinder an easy adaptation of the online learning, but the effect is expected to be less strong for representative elderly HA wearers.

In the sequence learning experiments, we wanted to improve the DRR performance by modeling the temporal correlation between neighboring samples to consider the order of the routine data. Here, we tested the **hidden Markov model** (HMM) with three classifiers as observation models and the long short-term memory (LSTM) network. We demonstrated that the HMM with the **multi-layer perceptron and random forest observation model** achieved the **best F-measure performance** on our set $\mathbb{D}_7$ and the Huynh set. Thereby, the MLP has the strongest F-measure improvement on both sets by adding the HMM. The LSTM has a worse F-measure performance on both sets. We analyzed the temporal behavior of prediction errors by defining segment errors and discovered that the sequence learners improve as expected the temporal stability of the predictions. Thereby, the overall errors are strongly reduced, and the predictions change less over time. However, this introduces new errors in some situations, where the predictions are too temporally stable. On our set, the remaining confusion for classifiers mainly stems from the intention-based class decision of the HA users.

7.2 Outlook

In this section, possible future investigations and improvements are outlined for the daily routine recognition. First of all, long-term studies are proposed with representative hearing aid users. Thereby, possible challenges, like a concept drift, and improvements, like an automatic labeling system, are discussed. Further potential investigations on an improved feature representation are explained with additional data analytics on the used features and new beneficial sensor modalities are suggested. Since hearing strongly depends on the situational intentions, possible solutions to decode the intentions are introduced. In addition, a potential application of DRR is explained with individual routine classes as well. Further algorithmic enhancements are proposed to improve the daily routine recognition.

Long-term Studies with Representative Hearing Aid Wearers

In this dissertation, we performed our experiments on two data sets with non-representative subjects and an approximated duration of about two weeks per subject. Therefore, **long-term studies with representative HA wearers** should be carried out to determine the DRR performance based on these elderly people. We can investigate, if they perform other activities, have weaker motion patterns, or their general routine structure is different. Nevertheless, it is expected, that these elderly users rely more on a recurring set of activities and environments [79]. In general, old people are less socially active and mostly stay at home [7]. Thus, the daily routine recognition should be simplified for the elderly HA wearers.

Emerging Robustness Challenges of Long-term Studies

In case of long-term recordings for many weeks, a **concept drift** may change the routine distribution over time [61]. In our work so far, we only found a recurring drift, which happens due to the different behavior of weekends and weekdays, since more free time activities are performed at the weekend. In general, gradual behavior changes can happen due to, e.g., a starting illness, like dementia or depression [149, 150]. This could result in reduced physical activities or less talking in conversations. Alternatively, an abrupt change may be due to a broken leg, which results in a severe reduction of movements. Therefore, in future work, long-term recordings should test the system robustness against stronger routine distribution modifications.

Proposed Improvements for Long-term Studies

Since long-term recordings can be annoying for the subjects due to the constant burden to label the current situation, an **automatic labeling system** can strongly reduce the burden. Hence, a pretrained classification system could run in the smartphone and trigger a notification alarm on the mobile phone or directly as playback tone in the HA to remind subjects to label the current situation. In particular, a user feedback is necessary if the system is uncertain about the current situation, which can be assessed by the outputted score of a classifier. In some non-appropriate situations, e.g., a conversation, a subject does not want to be interrupted [27, 28].

Thus, the system can store the time point of change to allow a retrospective labeling when it is appropriate, again. This has the advantage that the natural behavior is not interrupted, and people do not usually know the exact time point of the situation change, but they remember the executed activities for some time. Thus, the quality of user feedback in terms of timing issues and forgotten annotations would be improved.

Binaural Motion Features

Besides using a two-stage statistical **feature representation**, further improvements can be made to enhance the description power of the features for the routine classes. Since the recordings of two hearing aids are not synchronous in time, we could not use a binaural processing of the acceleration signals for advanced head rotation features [84]. By temporally aligning both sides, the yaw rotation detection can particularly be enhanced, which could lead to information about the angular range of motion or viewing direction presence probability [73, 74]. With these inputs, conversational situations with a high yaw activity and working on a computer with less head movements could be better differentiated. In addition, the nodding behavior during conversations can be exploited more, since often a fast-nodding behavior from a listener is seen [45].

Investigations on other Sensors

Further sensors than the ACC sensor could be as well beneficial to retrieve information about the location, environmental parameters, physical activity, or biomedical signals. An overview of different sensor modalities can be found in [2]. For example, a built-in magnetometer in a HA allows to directly measure the absolute yaw angle deviation from the north direction. This would be an easier alternative to analyze the yaw rotation behavior instead of a more complex analysis of two accelerometers. A gyroscope would allow a direct measurement of the 3D angular velocity to detect all three possible head rotations. Furthermore, including environmental data can be helpful to gain more context information, for example, the location being at a restaurant, home, or work. This information can be retrieved in a smartphone from a GPS sensor, geofencing, WiFi network connection, ambient sensors, or cell towers [151, 152].

Investigations on Event Duration and Time of Day

In our work, the **time of day** feature had no successful contribution in section 4.2 on both of our data sets $\mathbb{D}_7$ and $\mathbb{D}_\mathrm{T}$, since our younger subjects had too much variability in the daily routine structure. For the Huynh set, a stronger beneficial effect was found, because his working routine was strongly structured, and the set had specific time-related classes like lunch. In general, the time of day feature may be beneficial for other subjects. In addition, from existing studies on the use of time, prior knowledge about a typical activity start time or **duration** can be incorporated [153]. In [154], an LSTM was used to predict the event duration, which could lead to a possible performance improvement. We also tested a duration model of an HMM

with a Gamma distribution, but we could not show an enhancement over a standard HMM [140], since the duration density is exponentially decaying in the set $\mathbb{D}_7$.

Intention based Daily Routine Classes

In our data set $\mathbb{D}_7$, we applied intention-based routine classes that the test subjects choose the appropriate class based on their preferences and intentions in this situation. This setup leads to an ambiguity in some complex situations, where multiple choices are possible. To better solve the **intention problem** in hearing or situation labeling, additional biomedical sensors can provide a more reliable input. For example, the listening effort correlates with complex hearing situations and produces stress for the body, which can be measured by the heart rate variability [155, 156]. In addition, electromyogram to measure the listening attention [148] or electroencephalography to decode brain activity [157] can distinguish the intended behavior more precisely. Suitable motion patterns, e.g., head rotations during conversations [2], can help as well to better decode the situational intentions. In our study with the seven subjects, we used generic intention-based classes to make our research comparable between the subjects, but instead **individual or group-based classes** could be applied [75]. Thus, each personal wish of a setting could be trained as one class. In this case, the preferences of different users may overlap. That is why, the wishes can be bundled on a group level for a target group to avoid a cold start of the system for new users with no stored personal wishes. The real-time application of full approach can be done in future with the integration of subject wishes and intentions. This would allow to measure the real audiological benefit for the representative hearing aid wearers. A possible integration in a hearing aid system with a proposed audiological setting optimization is described in the Annex A.2 and A.3.

Algorithmic Enhancements and Deep Learning Architectures

Further **algorithmic enhancements** on the classifiers can be made to increase the DRR performance. We showed the positive effect of online and sequence learning for the classification task. Obviously, both learning paradigms can be combined. Therefore, with **online sequence learning** the model parameters of an HMM would change over time, i.e., the transition matrix is adaptive and observation model is modified as well [158]. In addition, the decoding step needs to store the total sequence and finds an optimal class sequence. Therefore, an online Viterbi decoding algorithms should be tested to allow a real-time application [144, 142]. For the LSTM network, an online variant exists and can be tested [159]. In the online learning, we updated the model based on the true or predicted labels, but we could also update the online model based on the predicted labels and only ask for the true user labels if the model is uncertain. This could be measured by the output score of a classifier. **Further deep learning architectures** can be tested, such as convolutional or other recurrent neural networks like the gated recurrent unit [71, 72]. In another variant, the deep learning can be directly applied on the low-level acceleration and acoustic features, but it needs to be ensured that the network can either handle a variable sampling rate or a robust sampling scheme should stabilize the rate. This should improve the performance of the neural networks.

Handling of Class Imbalance and Multi-Label Problem

Furthermore, algorithms to better handle the **class imbalance** of the routine data sets should improve the results, because the frequency of different routine classes strongly varies across data sets and classes. In our case on the $\mathbb{D}_7$ set, we tested different sampling schemes to under-sample or upsample the instances of the minority or majority classes, but it did not improve our results. The same outcome was yielded by introducing different costs to weight the wrong predictions. Nevertheless, here lays some potential for a possible improvement [160, 161]. In addition, one-class learning for the support vector machine could be beneficial [162]. In our data set $\mathbb{D}_7$, some classes may occur at the same time and the user should select the intended dominant class, which is also called an overlapping activity. Alternatively, routine activities are started and then interrupted by another activity, but they are later on finished, which is called an interleaved activity. We formalized the classification task as a multi-class problem, i.e., one class is active at a time. Alternatively, the problem could be written as a **multi-label problem** [163] meaning multiple classes are active at the same time. This could better handle the overlapping routine activities. For interleaved events, a model having a longer temporal context may be beneficial. In addition, due to different intentions in similar situations, a local neighborhood weighting function could express the certainty of a class label [164], which is used as input to train a classifier in the multi-label problem.

Annex

In this chapter, the study protocol and manual to record the data set $\mathbb{D}_7$ is explained in section A.1. Furthermore, a solution is proposed to find ideal hearing aid settings based on the daily routine classes in sections A.2. In addition, two proposals are made on how to integrate the output of the daily routine classifier in an existing hearing aid with an acoustic classification system in section A.3.

A.1 Data Recording Study Protocol and Manual for the $\mathbb{D}_7$ set

To facilitate the recordings of the data set $\mathbb{D}_7$ for the subjects, the study protocol and manual was written in German and is shown in Fig. A.1. It explains the usage of the intention-based routine labels and the steps to follow before, during, and after the recording. The initially chosen class is miscellaneous (misc.), which stands for unlabeled instances and these intervals are deleted later on. For each of the other routine classes, the user has to manually select them, and a set of exemplary activities is given in the manual to facilitate a proper selection of the routine classes.

Before starting the recording, the hearing aids need to be once paired with the iPhone to allow the Bluetooth connection and the battery status needs to be checked.

During the recording, the user can check if the recording properly runs by an animation of the acceleration values and can restart a broken Bluetooth. Additionally, the subject can choose a routine label by selecting an option in the drop-down menu and should alter the choice if the routine class changes.

After the recording, the data are prepared for the export to the storage server. Therefore, the user inputs his or her initials plus the date for the reference ID of a file. A description of the data or missed class changes can be noted in the description field. After the successful upload, the data can be deleted from the smartphone.

A.2 Proposed Audiological Optimization of Personalized Routine Classes

After the daily routine recognition on the objective audio and acceleration data, a corresponding personalized device setting needs to be found and linked for each class as shown in Fig. A.2. Therefore, we propose to use the so-called sound sense learn approach of [165], which optimizes the HA parameters in a situation based on subjective A and B comparisons. Thus, the satisfaction of a user with its hearing aids strongly depends on the optimal choice of the processing parameters. That is why, the satisfaction is a function of the current device settings and the situational

Versuchsablauf

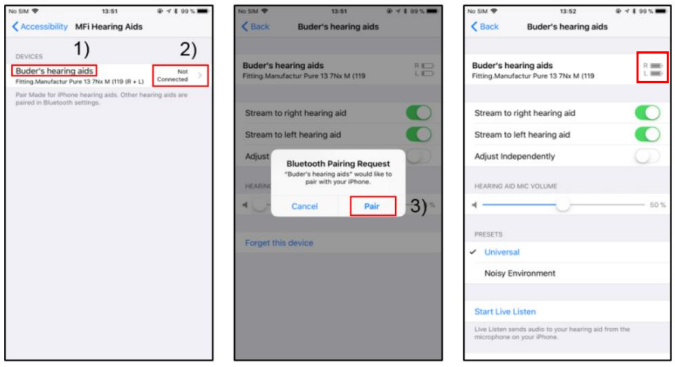
Erklärung der Labels:

Bitte die Zuordnung der Aktivitäten zu der Routine beachten und die Routine nach dem **aktuell beabsichtigten und dominanten Verhalten** wählen:
z. B: Gespräch während der Autofahrt → dominante Routine: Social

Routine	Activities
Transportation	Commuting, train, car, bus, plane, location change from A to B
Physical (Activity)	Physical exercises, workout, sport, manual work
Basics	Hygiene, dressing, resting, eating, preparing food, housekeeping, office work
Social (Interaction)	Family, friends, conversations, play music, singing, call
(Focused) Listening	Music, cinema, theatre, concert, lecture, TV, media
Misc.	Unregular activities, unknown/ unlabeled or does not fit

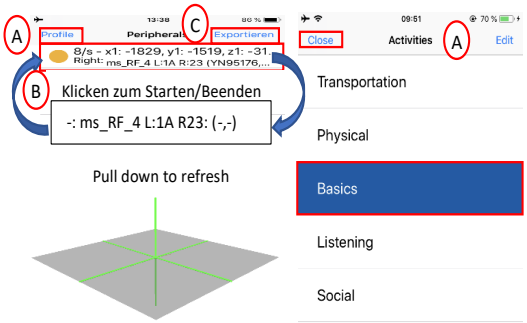
Vor der Aufnahme:

- HAs & Bluetooth aktivieren
- iPhone ggf. neustarten
- Unter "Settings" ⇒"General" ⇒ "Accessibility" ⇒ "MFi Hearing Aids"
  die **HA Verbindung zu iPhone** kontrollieren ggf. repairing **und Batteriestand kontrollieren** ggf. bei niedrigem Stand wechseln
- HAs einsetzen



Während der Aufnahme:

- Aktuelle Routine unter (A) Profile auswählen
- (B) Aufnahme starten durch auf HA/s klicken
- **Bei Änderung der Routine neues Label** unter Profile (A) vergeben
- **Regelmäßige Kontrolle der** bestehenden **Verbindung** von HA zum iPhone



Nach der Aufnahme:

(B) Aufnahme beenden und unter (C) exportieren gehen (**Bildschirm anlassen)**:

- (1) Daten exportieren
- (2) Reference-ID vergeben: Format: initialen_ttmmyyyy_last z. B: tk_05122018_last
- (3) Kurze Beschreibung geben: insbesondere, wenn außergewöhnliche (nicht typische oder „normale") Ereignisse auftreten z. B: Desk work, meeting with 5 people
- WLAN ggf. aktivieren, (4)-(9) Daten hochladen und beim erfolgreichen Upload Daten vom iPhone löschen

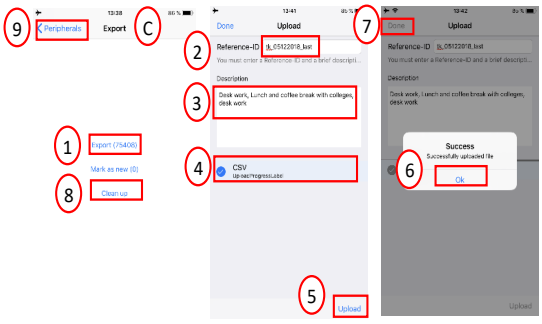


Figure A.1: The study protocol and manual for the subjects.

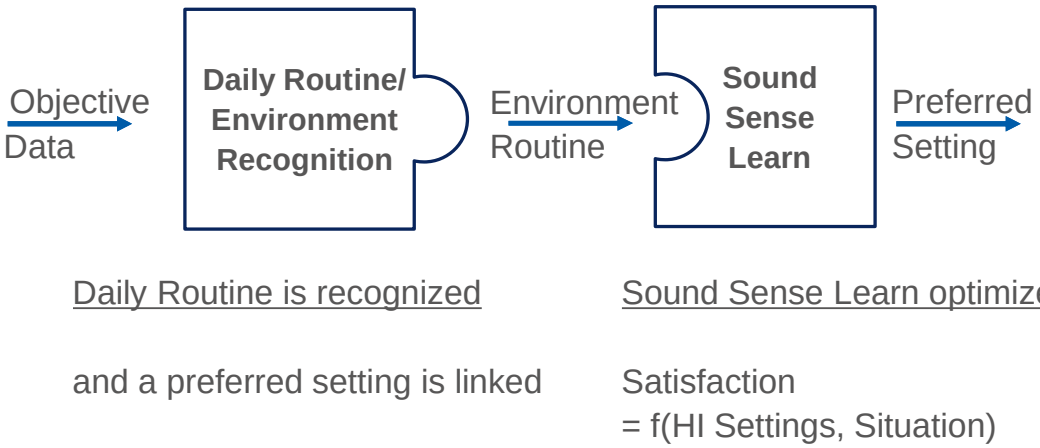


Figure A.2: The total approach for an improved and personalized hearing aid classification systems, where the daily routine classifier recognizes the current intention of a routine situation and, for example, the sound sense learn procedure finds a preferred hearing aid configuration for this situation.

characteristics. By comparing two audio processing options of two configurations A and B, the personal preference can be found out and the device settings are tuned to maximize the satisfaction. In that way, the hearing aid could be optimized for each daily routine class.

A.3 Integration of Personalized Daily Routine and Person-Independent Acoustic Classification

For now, a person-independent acoustic classification is performed in the hearing aids as stated in section 1.2. The Gaussian mixture model (GMM) is a popular classifier example in many audio or hearing aid studies [49, 64]. That is why, we use the GMM classifier as an example for our explanation. To integrate the personalized daily routine recognition into a hearing aid system, two methods are proposed by fusing the decisions in section A.3.1 or the settings in section A.3.2.

A.3.1 Fusion of Personalized and General Decision Making

The person-independent acoustic classification and the intention-based daily routine recognition can be combined as displayed in Fig. A.3 to fuse the decisions of both classification systems for an optimized decision making in terms of the temporal decision stability. Thus, the short-term acoustic decisions change in range of seconds, whereas the long-term routine recognition modifies its class decisions in the range of minutes to hours. Therefore, a balance between both systems achieves an overall temporal decision stability by fusing the routine and acoustic output probabilities to a combined score. This is done, for example, by Bayes averaging the scores, training a logistic regression, or a neural network on the combined score [166].

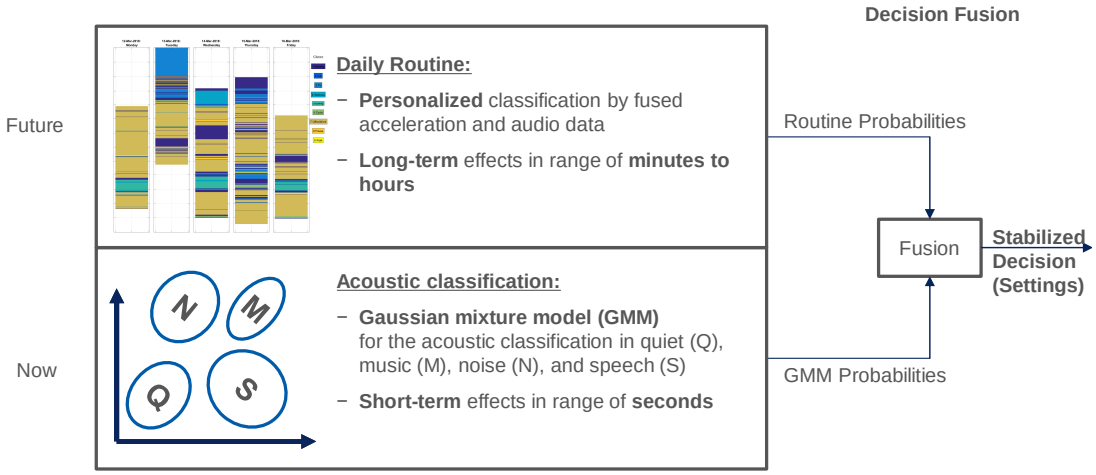


Figure A.3: The decision fusion of acoustic and daily routine classification, where the short-term acoustic decisions are balanced and stabilized by the slowly changing routine classification. The stabilized decision applies the optimized configuration.

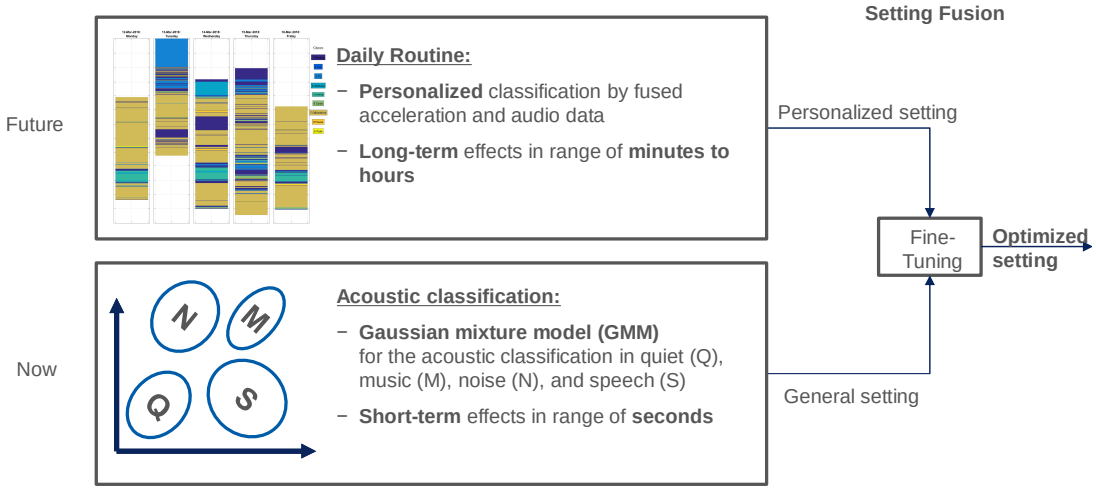


Figure A.4: The setting fusion of acoustic and daily routine classification, where the acoustic decision delivers a general setting, and the personalized routine classifier fine-tunes the configuration by the fusion of these two settings.

A.3.2 Fusion of Hearing Aid Settings from Person-Independent and Personalized Classification

In contrast to fuse the decisions, a fusion of settings is also possible in that way the acoustic classification provides a general setting and the recognized routine is linked with a personalized setting as shown in Fig. A.4. Thus, the routine classifier fine-tunes the settings of a generic acoustic classifier to an optimized preferred setting.

Abbreviations, Acronyms, and Symbols

ACC	Acceleration
ADL	Activities of Daily Life
AGNES	Agglomerative Nesting aka Hierarchical Clustering
cdf	Cumulative Distribution Function
CNN	Convolutional Neural Network
CV	Cross-Validation
dB	Decibel
DB	Davies Bouldin criterion
DBSCAN	Density-based Spatial Clustering of Applications with Noise
DFT	Discrete Fourier Transformation
DNN	Deep Neural Network
DR	Dimensionality Reduction
DRR	Daily Routine Recognition
DT	Decision Tree
EM	Expectation Maximization
EMA	Ecological Momentary Assessment
FFT	Fast Fourier Transformation
FN	False Negative
FNR	False Negative Rate
FP	False Positive
FPR	False Positive Rate
FS	Feature Selection
GMM	Gaussian Mixture Model
GT	Ground Truth
HA	Hearing Aid
HAR	Human Activity Recognition
HMM	Hidden Markov Model
Hz	Hertz
IMU	Inertial Measurement Unit
Isomap	Isomap
kNN	k-nearest neighbor
LDA	Linear Discriminant Analysis
LLE	Locally Linear Embedding
LODO	Leave-One-Day-Out
LOFO	Leave-One-Fold-Out
LOPO	Leave-One-Person-Out
LP	Label Propagation

LSTM	Long Short-term Memory
MAP	Maximum A Posteriori
MCR	Mean Crossing Rate
ML	Machine Learning
MLP	Multi-Layer Perceptron
NB	Naive Bayes
NF	Noise Floor
OVD	Own Voice Detection
ORF	Online Random Forest
p-value	Probability Value
pdf	Probability Density Function
P	Precision
R	Recall
RF	Random Forest
SC	Spectral Centroid
SD	Separation and Inter-cluster Distance Criterion
SFS	Sequential Feature Selection
Silh	Silhouette
SVD	Singular Value Decomposition
SVM	Support Vector Machine
std	Standard Deviation
TM	Topic Model
TN	True Negative
TP	True Positive
TPR	True Positive Rate
t-SNE	t-Distributed Stochastic Neighbor Embedding
UMAP	Uniform Manifold Approximation and Projection
VIL	Visual Interactive Labelling

α	angular acceleration
$\mathbf{\Sigma}$	covariance matrix
θ	angular position
λ	parameter vector or scalar
μ	mean
ρ	correlation coefficient
σ^2	variance
ω	angular velocity

A	Accuracy
a	acceleration
$\mathbf{a}$	acceleration vector
$\mathbf{a}_{\text{cal}}$	calibrated acceleration vector
$\mathbf{a}_{\text{dyn}}$	dynamic acceleration vector
$\mathbf{a}_{\text{lin}}$	linear acceleration vector
$\mathbf{a}_{\text{mes}}$	measured acceleration vector
$\mathbf{a}_{\text{R}}$	radial acceleration vector
$\mathbf{a}_{\text{T}}$	tangential acceleration vector

b	bias
c	a class or cluster c_i
ch	acoustic channel
$\mathbb{C}$	set of classes or clusters
$\mathbf{D}$	distance matrix
f	frequency
F_1	F-measure
g	gravitational acceleration of $9.81\frac{m}{s^2}$
i, j	indexes
K	number of classes or clusters
L	number of layers
l	label
$\mathbb{L}_p$	Minkowski norm of p
lvl	acoustic level
M	number of dimensions
N	number of samples
n	noise
$\mathbf{R}$	rotation matrix
r	radius
w	weight
x	feature attribute
$\mathbf{x}$	feature vector

List of References

[1] WHO, "Deafness and hearing loss." https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss, Mar. 2020.

[2] B. Tessendorf, *Multimodal Sensor and Actuator System for Hearing Instruments*. PhD thesis, ETH Zurich, 2012.

[3] H. Dillon, *Hearing Aids*. Boomerang Press, 2012.

[4] V. Hamacher, J. Chalupper, J. Eggers, E. Fischer, U. Kornagel, H. Puder, and U. Rass, "Signal processing in high-end hearing aids: State of the art, challenges, and future trends," *EURASIP Journal on Applied Signal Processing*, 2005.

[5] M. Büchler, S. Allegro, S. Launer, and N. Dillier, "Sound classification in hearing aids inspired by auditory scene analysis," *EURASIP Journal on Applied Signal Processing*, 2005.

[6] B. Tessendorf, A. Bulling, D. Roggen, T. Stiefmeier, M. Feilner, P. Derleth, and G. Tröster, "Recognition of hearing needs from body and eye movements to improve hearing instruments," in *International Conference on Pervasive Computing*, (Berlin, Heidelberg), pp. 314–331, Springer, 2011.

[7] A. L. Horgas, H.-U. Wilms, and M. M. Baltes, "Daily life in very old age: Everyday activities as expression of successful living," *The Gerontologist*, vol. 38, no. 5, pp. 556–568, 1998.

[8] W. English, "Daily routines - english vocabulary." `https://bit.ly/2Og7RMC`, Nov. 2019.

[9] T. Kuebert, H. Puder, and H. Koeppl, "Daily routine recognition with visual interactive labeling by fusing acceleration and audio signals," in *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 1–6, Dec. 2019.

[10] T. Kuebert, H. Puder, and H. Koeppl, "Daily routine recognition for hearing aid personalization," *SN Computer Science*, vol. 2, pp. 1–12, Mar. 2021.

[11] T. Kuebert, H. Puder, and H. Koeppl, "Improving daily routine recognition in hearing aids using sequence learning," *IEEE Access*, vol. 9, pp. 93237–93247, June 2021.

[12] O. Banos, J.-M. Galvez, M. Damas, H. Pomares, and I. Rojas, "Window size impact in human activity recognition," *Sensors*, vol. 14, pp. 6474–6499, Apr. 2014.

[13] M. Janidarmian, A. R. Fekr, K. Radecka, and Z. Zilic, "A comprehensive analysis on wearable acceleration sensors in human activity recognition," *Sensors*, 2017.

[14] E. De-La-Hoz-Franco, P. Ariza-Colpas, J. M. Quero, and M. Espinilla, "Sensor-based datasets for human activity recognition – a systematic review of literature," *IEEE Access*, vol. 6, pp. 59192–59210, 2018.

[15] R. Chavarriagaa, H. Saghaa, A. Calatronib, S. T. Digumartia, G. Tröster, J. del R. Millán, and D. Roggen, "The opportunity challenge: A benchmark database for on-body sensor-based activity recognition," *Pattern Recognition Letters*, vol. 34, pp. 2033–2042, 2013.

[16] J. Windau and L. Itti, "Walking compass with head-mounted imu sensor," in *IEEE International Conference on Robotics and Automation (ICRA) Stockholm, Sweden, May 16-21, 2016*, University of Southern California, IEEE, May 2016.

[17] T. Van Kasteren, G. Englebienne, and B. J. Kröse, "Activity recognition using semi-markov models on real world smart home datasets," *Journal of ambient intelligence and smart environments*, vol. 2, no. 3, pp. 311–325, 2010.

[18] T. Huynh, M. Fritz, and B. Schiele, "Discovery of activity patterns using topic models," in *Proceedings of the 10th International Conference on Ubiquitous Computing*, vol. 8, pp. 10–19, 2008.

[19] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, pp. 1733–1746, Oct. 2015.

[20] V. Zue, S. Seneff, and J. Glass, "Speech database development at mit: Timit and beyond," *Speech communication*, vol. 9, no. 4, pp. 351–356, 1990.

[21] Mozilla, "Common voice." https://www.kaggle.com/mozillaorg/common-voice/home.

[22] J. L. Roux and E. Vincent, "A categorization of robust speech processing datasets," in *Mitsubishi Electric Research Laboratories Technical Report, TR2014-116*, Aug. 2014.

[23] C. Dossman, "Over 1.5 tb's of labeled audio datasets," Nov. 2018.

[24] F. Seipel and C. Jaedicke, "Resources for audio event detection & scene analysis w/ml," *Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1088–1110, 2020.

[25] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, "Dcase 2017 challenge setup: Tasks, datasets and baseline system," in *DCASE 2017-Workshop on Detection and Classification of Acoustic Scenes and Events*, 2017.

[26] C. Lopes and F. Perdigão, "Phone recognition on the timit database," *Speech Technologies/Book*, vol. 1, pp. 285–302, 2011.

[27] P. von Gablenz, I. Holube, U. Kowalk, S. Bilert, M. Meis, and J. Bitzer, "Data analysis from real-world hearing assessment," in *International Hearing Aid Research Conference (IHCON), Lake Tahoe, CA, USA*, 2018.

[28] N. Schinkel-Bielefeld, P. Kunz, A. Zutz, E. Droste, and B. Buder, "Evaluation of hearing aids using ecological momentary assessment (EMA) - what situations are we missing?," *American Journal of Audiology*, pp. 591–609, 2020.

[29] D. Xu and Y. Tian, "A comprehensive survey of clustering algorithms," *Annals of Data Science*, vol. 2, pp. 165–193, June 2015.

[30] A. Stisen, H. Blunck, S. Bhattacharya, T. S. Prentow, M. B. Kjærgaard, A. Dey, T. Sonne, and M. M. Jensen, "Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition," in *13th ACM conference on embedded networked sensor systems*, pp. 127–140, 2015.

[31] C. Dobbins and R. Rawassizadeh, "Towards clustering of mobile and smartwatch accelerometer data for physical activity recognition," *Informatics*, vol. 5, p. 29, June 2018.

[32] L. G. Fahad, A. Ali, and M. Rajarajan, "Long term analysis of daily activities in a smart home," in *ESANN 2013 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Apr. 2013. Bruges (Belgium).

[33] D. A. Keim and H.-P. Kriegel, "Visualization techniques for mining large databases: A comparison," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, IEEE, Dec. 1996.

[34] D. A. Keim, "Information visualization and visual data mining," in *IEEE Transactions on Visualization and Computer Graphics*, vol. 7, IEEE, Feb. 2002.

[35] D. Marghescu, "Multidimensional data visualization techniques for financial performance data: A review," Tech. Rep. Report, Turku Centre for Computer Science, 2007.

[36] L. Van Der Maaten, E. Postma, and J. Van den Herik, "Dimensionality reduction: A comparative review," *J Mach Learn Res*, vol. 10, pp. 66–71, 2009.

[37] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," in *Journal of Machine Learning Research*, vol. 9, Nov. 2008.

[38] J. Bernard, M. Hutter, M. Zeppelzauer, D. Fellner, and M. Sedlmair, "Comparing visual-interactive labeling with active learning: An experimental study," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 298–308, 2018.

[39] B. Settles, "Active learning literature survey," Tech. Rep. 1648, University of Wisconsin–Madison, Jan. 2010.

[40] M. Stikic, D. Larlus, S. Ebert, and B. Schiele, "Weakly supervised recognition of daily life activities with wearable sensors," in *IEEE transactions on pattern analysis and machine intelligence*, pp. 2521–2537, 2011.

[41] A. Kumar and B. Raj, "Audio event detection using weakly labeled data," in *Proceedings of the 24th ACM international conference on Multimedia*, Language Technologies Institute, July 2016.

[42] A. Zinnen, U. Blanke, and B. Schiele, "An analysis of sensor-oriented vs. model-based activity recognition," in *IEEE International Symposium on Wearable Computers (ISWC)*, 2009.

[43] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[44] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surv.*, vol. 46, pp. 33:1–33:33, Jan. 2014.

[45] J. Hale, J. A. Ward, F. Buccheri, D. Oliver, and A. F. de C Hamilton, "Are you on my wavelength? interpersonal coordination in naturalistic conversations," *Journal of Nonverbal Behavior*, vol. 44, no. 1, 2018.

[46] S. Hemminki, P. Nurmi, and S. Tarkoma, "Accelerometer-based transportation mode detection on smartphones," in *Proceedings of the 11th ACM conference on embedded networked sensor systems*, 2013.

[47] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the cuidado project," tech. rep., Institute for Research and Coordination in Acoustics/Music, Paris, 2004.

[48] E. Scheirer and M. Slaneyy, "Construction and evaluation of a robust multifeature speech/music discriminator," in *1997 IEEE international conference on acoustics, speech, and signal processing*, vol. 2, pp. 1331–1334, IEEE, 1997.

[49] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1733–1746, 2015.

[50] J. Schroeder, S. Wabnik, P. W. van Hengel, and S. Goetze, "Detection and classification of acoustic events for in-home care," in *Ambient assisted living*, pp. 181–195, Berlin, Heidelberg: Springer, 2011.

[51] A. Temko, *Acoustic Event Detection and Classification.* PhD thesis, Department of Signal Theory and Communications, Universitat Politècnica de Catalunya, 2007.

[52] B. Fu, N. Damer, F. Kirchbuchner, and A. Kuijper, "Sensing technology for human activity recognition: A comprehensive survey," *IEEE Access*, vol. 8, pp. 83791–83820, 2020.

[53] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, "Activity classification using realistic data from wearable sensors," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, pp. 119–128, Jan. 2006.

[54] P. Lukowicz, J. A. Ward, H. Junker, M. Stäger, G. Tröster, A. Atrash, and T. Starner, "Recognizing workshop activity using body worn microphones and accelerometers," *International conference on pervasive computing*, pp. 18–32, 2004.

[55] J. A. Ward, P. Lukowicz, G. Tröster, and T. E. Starner, "Activity recognition of assembly tasks using body-worn microphones and accelerometers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1553–1567, Oct. 2006.

[56] U. Blanke and B. Schiele, "Daily routine recognition through activity spotting," in *Lecture Notes in Computer Science*, pp. 192–206, Springer Berlin Heidelberg, 2009.

[57] T. Huynh, *Human Activity Recognition with Wearable Sensors*. PhD thesis, TU Darmstadt, 2008.

[58] J. Seiter, *Topic Models for Activity Discovery in Daily Life*. PhD thesis, ETH Zurich, 2015.

[59] R. J. White, *Using Topic Models to Detect Behaviour Patterns for Healthcare Monitoring*. PhD thesis, University of Reading, Systems Engineering, 2018.

[60] F. Sun, Y.-T. Yeh, H. Cheng, C. Kuo, and M. Griss, "Nonparametric discovery of human routines from sensor data," in *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pp. 11–19, 2014.

[61] G. Ditzler, M. Roveri, C. Alippi, and R. Polikar, "Learning in nonstationary environments: A survey," *IEEE Computational Intelligence Magazine*, vol. 10, pp. 12–25, Nov. 2015.

[62] P. Laskov, C. Gehl, S. Krueger, and K.-R. Mueller, "Incremental support vector learning: Analysis, implementation and applications," *Journal of Machine Learning Research 7*, pp. 1909–1936, 2006.

[63] V. Losing, B. Hammer, and H. Wersing, "Choosing the best algorithm for an incremental on-line learning task," in *European Symposium on Artificial Neural Networks*, 2016.

[64] M. R. Schädler and B. Kollmeier, "Separable spectro-temporal Gabor filter bank features: Reducing the complexity of robust features for automatic speech recognition," *Acoustical Society of America*, 2015.

[65] P. Siirtola, H. Koskimaki, and J. Roning, "Personalizing human activity recognition models using incremental learning," in *ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, (Bruges (Belgium)), Apr. 2018.

[66] T. Sztyler and H. Stuckenschmidt, "Online personalization of cross-subjects based activity recognition models on wearable devices," in *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, IEEE, 2017.

[67] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," *Information Fusion*, vol. 37, pp. 132–156, 2017.

[68] T. G. Dietterich, "Machine learning for sequential data: A review," in *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*, pp. 15–30, Springer, 2002.

[69] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[70] S. Mota, R. W. Picard, M. Media, and Laboratory, "Automated posture analysis for detecting learner's interest level," in *Conf. Comput. Vis. Pattern Recognit. Workshop*, 2003.

[71] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, no. 3 – 11, 2019.

[72] N. Y. Hammerla, S. Halloran, and T. Plötz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 2016.

[73] T. Kuebert and T. Wurzbacher, "Method for operating a hearing aid, and hearing aid," United States Patent Application US20210051420A1, Feb. 2021.

[74] T. Wurzbacher, T. Kuebert, and D. Mauler, "Method for operating a hearing device and hearing device," United States Patent US010959028B2, Mar. 2021.

[75] T. Kuebert and S. Aschoff, "Method for the environment-dependent operation of a hearing system and hearing system," United States Patent Application US20210176572A1, June 2021.

[76] VRguy, "What you should know about head trackers." `http://vrguy.blogspot.de/2013/05/what-you-should-know-about-head-trackers.html`, May 2013.

[77] "Google picture search: Phd hut." `https://www.google.de/imghp?hl=de`, Oct. 2020.

[78] S. S. Hasan, O. Chipara, and Y.-H. Wu, "Evaluating auditory contexts and their impacts on hearing aid outcomes with mobile phones," in *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare*, pp. 126–133, 2014.

[79] Y.-H. Wu and R. A. Bentler, "Do older adults have social lifestyles that place fewer demands on hearing?," in *Journal of the American Academy of Audiology*, vol. 23, pp. 697–711, 2012.

[80] F. Wolters, K. Smeds, E. Schmidt, E. K. Christensen, and C. Norup, "Common sound scenarios: A context-driven categorization of everyday sound environments for application in hearing-device research," *Journal of the American Academy of Audiology*, vol. 27, pp. 527–540, July 2016.

[81] A. S. Bregman, *Auditory scene analysis: The perceptual organization of sound*. MIT press, 1994.

[82] M. C. Büchler, *Algorithms for Sound Classification in Hearing Instruments*. PhD thesis, ETH Zurich, 2002.

[83] C. Debes, A. Merentitis, S. Sukhanov, M. Niessen, N. Frangiadakis, and A. Bauer, "Monitoring activities of daily living in smart homes: Understanding human behavior," *IEEE Signal Processing Magazine*, vol. 33, pp. 81–94, Mar. 2016.

[84] T. Kuebert, "Classification of head and body movements using acceleration sensors in hearing aids," Master's thesis, Department of Electrical Engineering and Information Technology, Technical University of Darmstadt, 2016.

[85] M. Saar-Tsechansky and F. Provost, "Handling missing values when applying classification models," *Journal of Machine Learning Research 8*, pp. 1625–1657, 2007.

[86] O. P. John and S. Srivastava, "The big five trait taxonomy: History, measurement, and theoretical perspectives," *Handbook of personality: Theory and research*, vol. 2, pp. 102–138, 1999.

[87] P. Schopp, L. Klingbeil, C. Peters, A. Buhmann, and Y. Manoli, "Sensor fusion algorithm and calibration for a gyroscope-free IMU," *Procedia Chemistry*, vol. 1, pp. 1323–1326, Sept. 2009.

[88] I. Georgieva, C. Hofreither, T. Ilieva, T. Ivanov, and S. Nakov, "Laboratory calibration of a mems and accelerometer sensor," 1995.

[89] A. Cirillo, P. Cirillo, G. De Maria, C. Natale, and S. Pirozzi, "A comparison of multisensor attitude and estimation algorithms," Sept. 2016.

[90] T. Powers, M. Froehlich, E. Branda, and J. Weber, "Clinical study shows significant benefit of own voice processing," tech. rep., Hearing Review, 2018.

[91] M. F. McKinney and J. Breebaart, "Features for audio and music classification," in *Proceedings of the Third International Symposium on Music Information Retrieval*, pp. 151 – 158, 2003.

[92] T. Huynh, U. Blanke, and B. Schiele, "Scalable recognition of daily activities with wearable sensors," in *International Symposium on Location-and Context-Awareness*, pp. 50–67, Springer, 2007.

[93] T. O. D. Beéck, W. Meert, K. Schütte, B. Vanwanseele, and J. Davis, "Fatigue prediction in outdoor runners via machine learning and sensor fusion," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD18*, pp. 606–615, ACM Press, 2018.

[94] J. A. Hartigan and M. A. Wong, "A k-means clustering algorithm," in *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 28, pp. 100–108, Wiley Online Library, 1979.

[95] D. Arthur, Sergei, and Vassilvitskii, "k-means++: The advantages of careful seeding," tech. rep., Stanford, 2006.

[96] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: an overview," in *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, pp. 86–97, Dec. 2012.

[97] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. L. Roux, and M. Ouimet, "Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering," *Advances in neural information processing systems*, vol. 16, pp. 177–184, 2003.

[98] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Kdd*, pp. 226–231, 1996.

[99] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, pp. 603–619, 2002.

[100] W. Wang, J. Yang, and R. Muntz, "STING: A statistical information grid approach to spatial data mining," in *VLDB*, vol. 97, pp. 186–195, 1997.

[101] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *National Science Review*, vol. 5, pp. 44–53, 2018.

[102] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *IEEE International Conference on Data Mining*, IEEE, Dec. 2010.

[103] C. Cassisi, P. Montalto, M. Aliotta, A. Cannata, and A. Pulvirenti, "Similarity measures and dimensionality reduction techniques for time series data mining," in *Advances in Data Mining Knowledge Discovery and Applications*, Sept. 2012.

[104] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, 1987.

[105] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE transactions on pattern analysis and machine intelligence*, no. 2, pp. 224–227, 1979.

[106] M. Halkidi, M. Vazirgiannis, and Y. Batistakis, "Quality scheme assessment in the clustering process," *PKDD*, pp. 265–276, 2000.

[107] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *International conference on database theory*, 2001.

[108] M. Wattenberg, F. Viégas, and I. Johnson, "How to use t-SNE effectively," *Distill*, 2016.

[109] D. Kobak and G. C. Linderman, "Umap does not preserve global structure any better than t-sne when using the same initialization," *bioRxiv*, 2019.

[110] K. Bunte, M. Biehl, and B. Hammer, "A general framework for dimensionality reducing data visualization mapping," *Neural Computation*, vol. 24, no. 3, pp. 771–804, 2012.

[111] J. W. Sammon, "A nonlinear mapping for data structure analysis," in *IEEE Transactions on computers*, vol. 100, pp. 401–409, IEEE, 1969.

[112] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *ArXiv e-prints 1802.03426*, 2018.

[113] A. M. Martinez and A. C. Kak, "PCA versus LDA," *IEEE transactions on pattern analysis and machine intelligence*, vol. 23, pp. 228–233, Feb. 2001.

[114] N. Y. Hammerla and T. Plötz, "Let's (not) stick together: Pairwise similarity biases cross-validation in activity recognition," *UBICOMP*, 2015.

[115] L. Breiman, "Random forests," in *Mach. Learn.*, vol. 45, pp. 5–32, Springer, 2001.

[116] D. Heylen, "Challenges ahead: Head movements and other social acts in conversations," *Virtual Social Agents*, pp. 45–52, 2005.

[117] A. Paxton and R. Dale, "Interpersonal movement synchrony responds to high- and low-level conversational constraints," *Frontiers in Psychology*, vol. 8, Jul 2017.

[118] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, pp. 156–167, Jan. 2006.

[119] S. Kecanovic, "Detection of communication scenarios based on accelerometer sensor data in hearing instruments," Master's thesis, Department of Electrical Engineering and Information Technology, Technical University of Darmstadt, 2018.

[120] C. Daube, "Detektion von Kopfgesten anhand von Beschleunigungssensoren mittels supervised-learning," Master's thesis, Lehrstuhl für Mustererkennung Department Informatik Friedrich-Alexander-Universität Erlangen-Nürnberg, 2020.

[121] J. Tang, S. Alelyani, and H. Liu, "Feature selection for classification: A review," *Data classification: Algorithms and applications*, 2014.

[122] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, pp. 1226–1238, 2005.

[123] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, no. Mar., pp. 1157–1182, 2003.

[124] H. Zhang, "The optimality of naive Bayes," *American Association for Artificial Intelligence*, vol. 1, no. 2, p. 3, 2004.

[125] P. Ahrendt, "The multivariate gaussian probability distribution," *Technical University of Denmark, Tech. Rep*, no. Denmark, 2005.

[126] A. Criminisi, J. Shotton, and E. Konukoglu, "Decision forests for classication, regression, density estimation, manifold learning and semi-supervised learning," *Microsoft Research technical report*, 2011.

[127] M. Wainberg, B. Alipanahi, and B. J. Frey, "Are random forests truly the best classifiers?," *Journal of Machine Learning Research 17*, 2016.

[128] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International Conference on Artificial Intelligence and Statistics*, 2010.

[129] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[130] O. Chapelle, P. Haffner, and V. N. Vapnik, "Support vector machines for histogram-based image classification," *IEEE Tranctions on Neural Networks*, vol. 10, pp. 1055–1064, Sept. 1999.

[131] A. Bifet, G. de Francisci Morales, J. Read, G. Holmes, and B. Pfahringer, "Efficient online evaluation of big data stream classifiers," *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 59–68, 2015.

[132] J. D. M. Rennie, L. Shih, J. Teevan, and D. R. Karger, "Tackling the poor assumptions of naive bayes text classifiers," in *Proceedings of the 20th international conference on machine learning (ICML-03)*, Artificial Intelligence Laboratory; Massachusetts Institute of Technology; Cambridge, 2003.

[133] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Editorial: Special issue on learning from imbalanced data sets," *ACM SIGKDD explorations newsletter*, vol. 6, pp. 1–6, 2004.

[134] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[135] T. F. Chan, G. H. Golub, and R. J. LeVeque, "Updating formulae and a pairwise algorithm for computing sample variances," in *COMPSTAT 1982 5th Symposium held at Toulouse*, pp. 30–41, Springer, 1982.

[136] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE transactions on speech and audio processing*, vol. 3, no. 1, pp. 72–83, 1995.

[137] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT 2010*, pp. 177–186, Springer, 2010.

[138] E. M. Knorr and R. T. Ng, "A unified notion of outliers: Properties and computation," in *KDD*, pp. 219–222, 1997.

[139] P. Siirtola and J. Röning, "Incremental learning to personalize human activity recognition models: The importance of human AI collaboration," *Sensors*, vol. 19, p. 5151, Nov. 2019.

[140] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257–286, 1989.

[141] C. M. Bishop, M. Svensen, and G. E. Hinton, "Distinguishing text from graphics in on-line handwritten ink," in *International Workshop on Frontiers in Handwriting Recognition (IWFHR)*, IEEE, 2004.

[142] G. Wang and R. Zimmermann, "Eddy: An error-bounded delay-bounded real-time map matching algorithm using HMM and online viterbi decoder," in *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 33–42, 2014.

[143] A. Churbanov and S. Winters-Hilt, "Implementing EM and viterbi algorithms for hidden markov model in linear memory," *BMC Bioinformatics*, vol. 9, Apr. 2008.

[144] J. Bloit and X. Rodet, "Short-time viterbi for online HMM decoding: Evaluation on a real-time phone recognition task," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008.

[145] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[146] G. Forman and M. Scholz, "Apples-to-apples in cross-validation studies: Pitfalls in classifier performance measurement," *ACM SIGKDD Explorations Newsletter*, 2009.

[147] J. A. Ward, P. Lukowicz, and H. W. Gellersen, "Performance metrics for activity recognition," *ACM Trans. Intell. Syst. Technol.*, vol. 2, pp. 6:1–6:23, Jan. 2011.

[148] D. J. Strauss, F. I. Corona-Strauss, A. Schroeer, P. Flotho, R. Hannemann, and S. A. Hackley, "Vestigial auriculomotor activity indicates the direction of auditory attention in humans," *eLife*, vol. 9, 2020.

[149] R. F. Dickerson, E. I. Gorlin, and J. A. Stankovic, "Empath: a continuous remote emotional health monitoring system for depressive illness," in *Proceedings of the 2nd Conference on Wireless Health*, pp. 1–10, 2011.

[150] A. Barua, A. K. M. Masum, E. H. Bahadur, M. R. Alam, M. A. U. Z. Chowdhury, and M. S. Alam, "Human activity recognition in prognosis of depression using long short-term memory approach," *International Journal of Advanced Science and Technology*, 2020.

[151] K. Farrahi and D. Gatica-Perez, "Discovering routines from large-scale human locations using probabilistic topic models," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 1, pp. 1–27, 2011.

[152] Y. Wang, S. Cang, and H. Yu, "A data fusion-based hybrid sensory system for older people's daily activity and daily routine recognition," *IEEE Sensors Journal*, vol. 18, no. 16, pp. 6874–6888, 2018.

[153] K. Partridge and P. Golle, "On using existing time-use study data for ubiquitous computing applications," in *UbiComp*, 2008.

[154] K. Krishna, D. Jain, S. V. Mehta, and S. Choudhary, "An LSTM based system for prediction of human activities with durations," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, pp. 1–31, Jan. 2018.

[155] S. Alhanbali, P. Dawes, R. E. Millman, and K. J. Munro, "Measures of listening effort are multidimensional," *Ear and Hearing*, vol. 40, no. 5, 2019.

[156] J. Taelman, S. Vandeput, A. Spaepen, and S. Van Huffel, "Influence of mental stress on heart rate and heart rate variability," in *4th European conference of the international federation for medical and biological engineering*, pp. 1366–1369, Springer, 2009.

[157] Y. Zheng, X. Ding, C. C. Y. Poon, B. P. L. Lo, H. Zhang, X. Zhou, G. Yang, N. Zhao, and Y. Zhang, "Unobtrusive sensing and wearable devices for health informatics," *IEEE Transactions on Biomedical Engineering*, vol. 61, pp. 1538–1554, 2014.

[158] T. Chis and P. G. Harrison, "Adapting hidden markov models for online learning," *Electronic Notes in Theoretical Computer Science*, vol. 318, pp. 109–127, Nov. 2015.

[159] T. Ergen and S. S. Kozat, "Efficient online learning algorithms based on LSTM neural networks," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 8, pp. 3772–3783, 2017.

[160] N. V. Chawla, "Data mining for imbalanced datasets: An overview," in *Data mining and knowledge discovery handbook*, pp. 875–886, 2009.

[161] A. Fernandez, S. Garca, F. Herrera, and N. V. Chawla, "Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary," *Journal of Artificial Intelligence Research 61*, pp. 863–905, 2018.

[162] R. Batuwita and V. Palade, "Class imbalance learning methods for support vector machines," *Imbalanced Learning: Foundations, Algorithms, and Applications; Wiley: Hoboken, NJ, USA*, 2013.

[163] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *International Journal of Data Warehousing and Mining (IJDWM)*, vol. 3, no. 3, 2007.

[164] Z. Qin, *Learning with Fuzzy Labels: A Random Set Approach Towards Intelligent Data Mining Systems*. PhD thesis, University of Bristol, 2005.

[165] N. S. Jensen, L. W. Balling, and J. B. Nielsen, "Effects of personalizing hearing-aid parameter settings using a real-time machine-learning approach," in *Proceedings of the 23rd International Congress on Acoustics*, (Aachen, Germany), Sept. 2019.

[166] D. Ruta and B. Gabrys, "An overview of classifier fusion methods," *Computing and Information Systems*, vol. 7, no. 1, pp. 1–10, 2000.