# Empathy for the User

## Towards True Peerlike Assistance Systems

**DISSERTATION**
zur Erlangung des akademischen Grades

**Doktoringenieur (Dr.-Ing.)**

von Norman Weisskirchen, M.Sc.
geb. am 15.12.1989 in Engelskirchen

genehmigt durch die
Fakultät für Elektrotechnik und Informationstechnik
der Otto-von-Guericke-Universität Magdeburg

Gutachter:
Prof. Dr. rer. nat. Andreas Wendemuth
Prof. Dr.-Ing. Christian Diedrich
Prof. Dr. Oliver Niebuhr

Promotionskolloquium am 27.06.2025

# Abstract

ONE of the most prevalent signs of integrated human-machine living is the wide array of available personal computer devices and their use as assistance systems. Over time this developed from fairly limited virtual notepads to smart-home integrated control and coordination tools adapted for their individual users. While this development increased both the value of the provided assistance, as well as the ease of accessibility for the individual user, with their specific needs and idiosyncrasies, it also developed towards specific limits hindering potential future developments and possible improvements. Given the current designs, the main area of research is concentrated on the optimization of human-machine interaction through easier and more robust voice control schemes to the detriment of true systematic improvements.

The original idea behind assistive technology ranges from the currently employed human-machine interfaces to much more integrated systems, which can be described as technical assistants, companions or even peers. All these expressions are intended to describe continuously closer approximations to human-like capabilities and responsibilities. The overarching topic of this thesis is an exploration of the different aspects of such an advanced assistive system, specifically how it connects with my research and how potential further research might be necessary for the fulfilment of this goal of a human-like technical assistant system.

The structural organisation of this thesis follows a potential information processing pipeline, as it could be used in such an assistance system, where the external inputs need to be extracted, categorised and interpreted depending on the current situation for a system to provide the necessary and adequate reactions, as required by the user. An important aspect of this will be the change from a purely user controlled reactive system, as usual nowadays, into a more independently deciding and acting system which is approximating human decision processes. The different areas of my research in relation to this pipeline are as follows:

The first area is the external feature layer. While this includes the whole technical aspect of interfacing the real world environment with the following technical system, in conjunction with my research it primarily contains examinations of the feature extraction and optimisation process. This is necessary, as a reasonably extensive and advanced system would work on a wide variety of information and input signals, leading to a fast approaching overload of most available personal

computer systems. A sensible pre-processing step can reduce the computational load and improve the results of the following calculations.

The second layer described is the categorisation layer of the system, practically where the purely technical feature values are categorised into assistance relevant information or classes. As a technical system possesses no human understanding or empathy, such abilities need to be approximated through other means. This part of my research deals with solutions using machine learning architectures as a base. Given the complexity and variability of human expressions, which can be strongly influenced by a high degree of individuality and dependency on the situation, a purely rule-based approach would reach its limits fast. Using self-learning methods, which employ human annotated examples instead, allows for a much easier implementation of non-directly measurable information into a system. Special attention is given to the effect of constantly increasing complex architectures and their requirements for equally increasing example data sets. A secondary topic examined here will be the search for alternatives to this paradigm of development through the use of less complex architectures without lowering the capabilities.

The third and last layer examined will be the decision making aspect, which proposes, based on the information from the former layers, a sensible action to follow. This part of the thesis will consist of two parallel examinations: First of a practical implementation of a semi-reactive method for working under uncertain situations, and second an examination into the theoretical implications of a cognitive architecture as full alternative for a control mechanism. This second example is a potential way into a human peer-like system architecture, which closely resembles the human decision making process and would allow an even closer integration into the human-machine environment than alternative current architectures.

# Zusammenfassung

EINES der am weitesten verbreiteten Anzeichen des integrierten Mensch-Maschine-Lebens ist die breite Palette verfügbarer Personal-Computer Geräte und deren Einsatz als Assistenzsysteme. Im Laufe der Zeit hat sich dies von relativ begrenzten virtuellen Notizblöcken zu Smart-Home integrierten Steuerungs- und Koordinationstools entwickelt, die an die jeweiligen Benutzer angepasst sind. Während diese Entwicklung sowohl den Gegenwert der bereitgestellten Hilfestellungen als auch die Verfügbarkeit für den Einzelnen, mit jeweils spezifischen Bedürfnissen und Eigenheiten, steigerte, entwickelte es sich auch in die Richtung spezifischer Grenzen die weitere potenzielle Entwicklungen und mögliche Verbesserungen verhindern. Angesichts der aktuellen Designs ist der Hauptforschungsbereich die Optimierung der Mensch-Maschinen-Interaktion durch einfachere und robustere Sprachsteuerungsmethoden, was zu Lasten echter systematischer Verbesserungen geht.

Die ursprüngliche Idee hinter Assistenztechnologien reicht von den derzeit verwendeten Mensch-Maschine-Schnittstellen bis hin zu viel stärker integrierten Systemen die mit Begriffen wie technischen Assistenten, Companions oder sogar Peers beschrieben werden können. All diese Ausdrücke sollen eine immer stärkere Annäherung an menschliche Fähigkeiten und Verantwortlichkeiten beschreiben. Das übergreifende Thema dieser Dissertation sind die Untersuchungen der verschiedenen Aspekte eines solchen fortschrittlichen Assistenzsystemes, speziell in Verbindung mit der von mir betriebenen Forschung. Potentielle weitere Entwicklungen werden diskutiert, um dieses Ziel eines menschenähnlichen technischen Assistenzsystems zu erreichen.

Die strukturelle Aufteilung in dieser Arbeit folgt einer Informationsverarbeitungspipeline, wie sie in einem solchen potentiellen Assistenzsystem verwendet werden könnte, wobei die externen Eingaben je nach aktueller Situation extrahiert, klassifiziert und interpretiert werden müssen, damit ein System das Notwendige und angemessenen Reaktionen, wie vom Benutzer gefordert, bereitstellen kann. Ein wichtiger Aspekt ist dabei der Wandel von einem rein benutzergesteuerten reaktiven System, wie es standardisiert üblich ist, zu einem eigenständiger entscheidenden und handelnden System, das sich menschlichen Entscheidungsprozessen annähert. Die verschiedenen Forschungsbereiche, die ich in Bezug auf diese Pipeline durchgeführt habe, sind wie folgt:

Der erste Bereich ist die externe Feature-Schicht. Diese umfasst übergreifend den

gesamten technischen Aspekt der Anbindung der realen Umgebung an die folgenden technischen Systeme, in Verbindung mit meiner Forschung aber vor allem Untersuchungen zu Merkmalsextraktionen und Optimierungsprozessen. Dies ist notwendig, da ein einigermaßen umfangreiches und fortschrittliches System mit einer Vielzahl von Informationen und Eingangssignalen arbeiten müsste, was zu einer raschen Überlastung der meisten verfügbaren persönlichen Computer Systeme führen würde. Ein sinnvoller Vorverarbeitungsschritt kann die Rechenlast verringern und die Ergebnisse der nachfolgenden Berechnungen verbessern.

Die zweite beschriebene Schicht ist die Interpretationsschicht des Systems, im praktischen dazu da, die rein technischen Merkmale in assistenzrelevante Informationen umzuwandeln. Da technische Systeme kein menschliches Verständnis oder Einfühlungsvermögen besitzen, müssen solche Fähigkeiten auf andere Weise erzeugt werden. In Anbetracht meines Forschungsgebiets befasst sich dieser Abschnitt mit Lösungen, die maschinelles Lernen als Grundlage verwenden. Angesichts der Komplexität und Variabilität der menschlichen Ausdrucksweise, speziell wenn beeinflusst durch ein hohes Maß an individuellen sprachlichen Gewohnheiten, sind selbst bei thematisch ähnlichen Äußerungen hohe Unterschiede festzustellen. Daher wird ein rein regelbasierter Ansatz schnell an seine technischen Grenzen stoßen müssen. Stattdessen können selbstlernende Methoden verwendet werden, die von Menschen annotierte Beispiele verwenden, damit nicht direkt messbare Informationen viel einfacher in ein System implementiert werden können. Besonderes Augenmerk wird dabei auf die Auswirkung immer komplexer werdender Systeme und deren Anforderungen an ebenso wachsende Beispieldatenmengen gelegt. Ein sekundäres Thema dieser Arbeit wird die Suche nach Alternativen zu diesem Entwicklungsparadigma durch die Verwendung weniger komplexer Architekturen ohne Verringerung der Fähigkeiten sein.

Die dritte und letzte untersuchte Ebene ist die Entscheidungsebene, die auf der Grundlage der Informationen aus den vorherigen Ebenen eine sinnvolle Maßnahme vorschlägt und befolgt. Dieser Teil besteht aus zwei parallelen Untersuchungen in dieser Arbeit. Erstens einer praktischen Umsetzung einer semi-reaktiven Methode für das Arbeiten in unsicheren Situationen und zweitens eine Untersuchung, die sich mit den theoretischen Implikationen einer kognitiven Architektur als ein Kontrollmechanismus befasst. Dieser zweite Punkt ist ein möglicher Weg in eine menschenähnliche Systemarchitektur, die dem menschlichen Entscheidungsfindungsprozess sehr ähnlich ist und eine noch engere Integration in die Mensch-Maschine-Umgebung ermöglichen würde als andere alternative Architekturen.

# Contents

# List of Figures

# List of Tables

Chapter 1

# Introduction

## Contents



Assistance Systems are an increasingly prevalent aspect of current Human-Machine Environments. The idea behind this technology is to introduce the wide array of technical possibilities into the everyday life of a human user and to improve through this the efficiency, safety and ease of use of all possible computer assisted activities. This concept of technological assistance can be followed from the first development of the personal computer systems till the current day use of mobile personal digital assistants. Through these ongoing developments, the general capabilities and integration of technical assistance systems were improved over time and became commercially easily available to greater parts of the population. At the same time, the variety of assistive capabilities from these systems became more constrained by the standardised designs which were implemented.

In this Thesis, I will explore the possibility of expanding the current approach for Human-Machine Interaction (HMI) based assistance systems. This proposed approach should be capable of processing a wide variety of real world information, as well as interpreting these based on collected human experiences through the use of machine learning architectures and finally be able to proceed with human-like reactions and decision making capabilities. These three aspects will be examined in greater detail in the following chapters, with connection to the exemplary research done by me for the different stages. The main part in this work will mainly examine acoustic and speech based input sources, based on the research done by me.

In this first chapter, the general structure and motivation of the thesis will be presented in detail. Specifically, in Section 1.1 the motivation for the proposed changes in the assistance system approach will be expounded. Section 1.2 goes then through each step of the general information pipeline, which is also the basis for the general structure of this work. This will describe the flow from the real world information gathering process into the technical system based decision making process and which problems occur along the way. In Section 1.3 these approaches and their problems will be connected specifically with the performed experiments done by me and then summarised with the aims and general research questions they impose. Finally, in Section 1.4 the internal structure of the thesis will be presented concerning the chapters and their functions in the overall examination of my research.

## 1.1 True Assistance Systems

The current level of assistance provided by technical systems, which are designated as personal assistance devices, is primarily the provision of mobile and easy access to technical systems. This includes database access, application control or similar technical interface functions. With these typical use cases for assistance systems, the focus of current development is mainly set on further optimisation of the accessibility and the implementation of additional applications and systems. Meanwhile the main functionality, as an easy HMI option is solved in most contemporary solutions through a form of voice control. These ongoing developments generally lack the potential to achieve a technical companion system, which should more closely resemble a human caretaker or colleague than a simple interface controller [Honold et al. 2014; Biundo & Wendemuth 2017].

To come closer to the proposed idea of an accompanying and helpful assistant, instead of the current HMI, one needs to examine the meaning of assistance or help in the context of a technical system. While a virtual assistant, as usual today, helps the user to access other applications, it is primarily designed as a reactive support. The user specifically declares which tasks are to be performed and then the system coordinates the necessary functions on this condition. The advantage of this is that all tasks are performed by the same overhead system, with a faster access compared to using the different applications individually by the user. The ideal assistant, as originally proposed, should instead perform the assistance proactively, in the sense of recognizing problems and solving them for the user as soon as possible, equal to a human assistant.

One of the main problems leading to this current state, which constricts most systems from performing as a real assistant and instead rely on direct user control, is the lack of understanding for the human condition. As a technical system has no underlying empathy behind their actions, it needs the human initiation to recognise a problem to solve. Without possessing a similar sense of self, as could only be achieved through currently futuristic solutions, such as true artificial intelligence, contemporary systems rely on rule-based approximations to control their reactions [Alty & Guida 1985]. A system possessing such a level of true human and emotional intelligence would most likely not be used as a personal assistance tool at all, because of the complex implications. But even then, a system would need to recognise inner human states, such as emotions or dispositions, to react with an appropriate action.

A result of this constraint is the inability of most systems to overcome their rule-based information processing structure. This allows most technical systems to only function in their predefined and preprogramed boundaries, in contrast to a real assistant, which would be able to adapt to new situations and occurrences [Müller 2011; Borst et al. 2022]. Even from the most advanced assistance system available, the underlying structure would require the system to wait for instructions from the user, as it would otherwise act in a non-planned and unknown way.

The aim of this thesis is to provide several different solutions for problems found in the current approach for an integrated human-machine environment, as well as possible optimisations to be done along the way. Finally, it will present an alternative overhead system approach, which would allow for a system to function in a more natural and proactive way compared to the usual reactive behaviour found in most of the currently available implementations, simulating a human-like behaviour without requiring true human intelligence from the system. The research done by me in this regard will be mainly focused on acoustic input and machine learning solutions to solve the problems mentioned.

The plan is to provide the basis for a system capable of approximating the human understanding and behaviour. For this the different parts of the information flow have to be approached separately. At first, it is necessary to improve the ability of the system to process the real world information gathered from the environment. While the amount and detail of available information can be increased by more and better sensors, it can lead to a so called "curse of dimensionality" problem, where the computational requirements increase exponentially with the available data [Bellman & Kalaba 1959]. Similar to human perception, where the concentration is aimed towards relevant changes, while most of the irrelevant data is ignored at least unconsciously, a system should be able to distinguish which received features contain the relevant aspects of information. Using this, the overarching flow of information can be increased without overloading the system with unnecessary noise.

Secondly, following this information gathering phase, the interpretation of the information is also an important area of potential improvement. Necessitating the interpretation of complex human mental and dispositional states, without natural human empathy, requires a classifier capable of inferring user states from externally measurable features. One solution for this kind of complex categorisation is the use of machine learning architectures. These can learn the correct correlations of features and states, based on annotated examples done by human experts on exemplary data. In this thesis, several improvements to the available methods will be explored, together with an examination of the advantages and disadvantages given by the available data sizes for different problems.

The third and final aspect is one of the biggest presented differences between a human-like system and a contemporary system, which is the difference between a proactive and a purely reactive user engagement. As a technical system cannot simply develop a sense of empathy but only an approximation, it also requires rules for engaging a user independently from direct inputs based commands. For this, the thesis provides an outlook on how to adapt a given system with a more proactive or at least semi-active mode, where the system may engage the user without direct former input required.

A closely connected aspect of this, which is examined in this thesis, is an overview on how to implement a human-like decision making structure on top of the information processing pipeline. This is also an important step towards a true assistance system, close to a technical companion, as it also includes the ability to produce its own set of directives and solutions during new and changing occurrences. This approach is labelled "peer-like" in my work, in the sense of a true peer during an interaction, similar or even equal to a human peer. In this thesis the first step towards this design will be examined theoretically as a possible implementation of a cognitive architecture based HMI.

The next section will closely examine these mentioned aspects of the research, which will also follow the general form of the overarching information flow structure going from the feature extraction stage, through the machine-learning based categorisation stage into the proactive and human-like decision making stage.

## 1.2 Peerlike Information Processing Pipeline

The general architecture of an assistance system can be understood as an elaborate information processing pipeline. Information need to be gathered, processed, interpreted and then reacted to by a control mechanism. Even a more advanced, proactive, system primarily reacts to a lack of information, which could not be gathered beforehand in the pipeline in conjunction of generating its own interpretations, similar to a human intelligence. This general structure is also the basis for the chapters in this thesis, with the separate layers of the process leading to the different topics and experiments for my research. As a short overview, this will be the data recording, feature extraction and pre-processing steps, which can be combined into one aspect of data gathering, followed by a larger topic of interpretation and categorisation of information in the middle layer and finally the controlling and reaction part at the top of the structure. This all can be seen in Figure 1.1. This structure can be seen as an amalgamation or improvement of the structure of a Driver Assistance System (DAS), which are exemplary for a close HMI environment, and the knowledge processing of companion systems. Specifically the typical perception/data- and decision/analytics- layers of DAS can be seen as fulfilling similar functions to the Sensor- and Interpreting–Layers respectively [Rezaei & Sabzevari 2009; Kohl et al. 2024], while the Cognitive Layer of my architecture more closely follows the knowledge processing of the companion architecture [Biundo & Wendemuth 2017]. Ideally this would mend the highly integrated nature of the one system, with the adaptability and versatility of the other one.

### 1.2.1 Peerlike Awareness

The first aspect of the described pipeline is the awareness. In the context of this thesis, and as part of a peer-like system, it primarily concerns itself with the ability of the system to process outside, i.e. real-world information into a machine

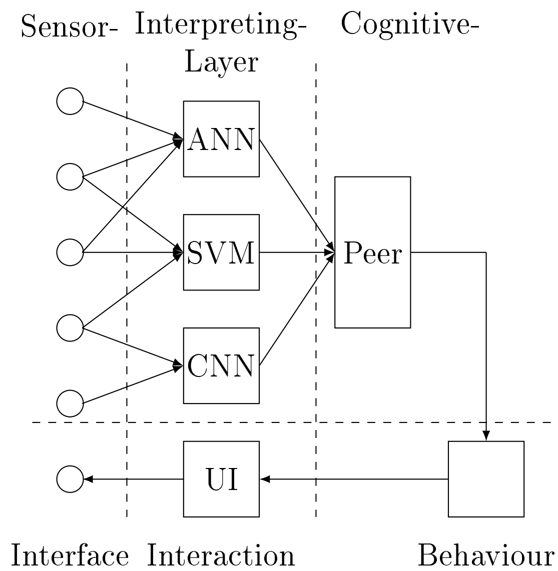



**Figure 1.1:** The fully extended data pipeline of the proposed peer-like Companion or Assistance System. Information from the real world gets transferred from the available sensors, after getting optimised, into the interpreting phase, using machine learning systems, in the second layer. The interpretations from there allow the final cognitive layer to choose the optimal actions based on former and newly generated knowledge, which is then communicated through the Behaviour control towards a User Interface back in the real world. The abbreviations in the Interpreting-Layer stand for different machine learning architectures.

interpretable format and to recognise significant and relevant changes for later classification tasks. Based on the presented structure before, this includes also the interface with which a user can interact with the system itself.

A great part of an aware system is the ability to discern environmental information. These contain primarily non-user dependent topics, such as outside influences. They can come from a wide range of available sensor information, like acoustic, visual or temperature sensors. In a typical system, these are employed as required by the primary function of the system, such as a temperature sensor for a smart-home heating application. In a peer-like system, which generates its context by correlating occurrence, even on the first view unrelated information can be important, leading to a more generalised information gathering approach. For a human-like generation of context, it is logical to provide sensory input based on the human senses. Vision and acoustic inputs are particularly important part in this context, both as an environmental, as well as an interpersonal source of information.

The other important part is the awareness for the state of the user, both their interest and their disposition. Using the aforementioned inputs these can often be determined by a technical system through direct statements by the user. For the currently typical approach, this is often straightforward, as the user actively explains their intention, requiring no further insight on side of the system. The current trend of wearable sensors, such as smart watches, would allow a technical system even further awareness beyond the typical human range, such as through bio-signals.

The combination of both environmental and user state awareness would allow a peer-like system to achieve situational understanding similar to a human, if it can achieve the correct conjunctions from them. While the interpretations and conclusions are part of the later aspects of the pipeline, the groundwork for this ability is taken from the available information gathered during this early stage. An important aspect is the required ability to not only discern the current information but also the dynamical changes happening over a certain time to recognize the flow from one situation to another.

The possible research in this area, specifically when concentrating on the digital part of this data flow, is the search for the most relevant aspects of the available data and the most significant features to reduce both the processing and storage requirements without limiting the learning capabilities of the system. The employed sensor systems in the real world are beyond the scope of this work, but would also allow for optimisation steps on this stage of the architecture.

With the resulting amount of available data most systems will sooner or later develop problems during the processing stage. While greater amounts of data principally correlate with better results during classification and decision making, the system has to be capable of ignoring superfluous features. While certain architectures and statistical procedures exist to assign different features with an evaluation of their information impact, it is often part of the design process from the human developer to find prior knowledge or empirical tests and to pre-choose the relevant features from the whole available dataset. Especially an independent architecture which is fully integrated into the life of their users will need to select its features and data sources carefully.

Generally this block remains close to the typical design of current assistance systems and follows the trend, that more available information generally improves the capabilities of the system itself, while at the same time decreases applicability with the available hardware.

### 1.2.2 Peerlike Understanding

The second aspect of the information pipeline is the information categorisation or classification step, which provides a technical system with context and understanding not directly extractable or measurable from the raw data. This is an important part of human to human communication, which is traditionally hard for a technical system to approach as it happens on a subconscious level not available to machines. Instead of a true artificial intelligence, which would provide for human-like sensibilities and understanding, the approximation is traditionally done through rule-based systems and typical user state classifications. These are often capable of working on emotion recognition tasks, through correlating sensor information with the otherwise hidden inner user states [Fragopanagos & Taylor 2005; Sapra et al. 2013].

In the context of human emotion or disposition, this can be done through a variety of multimodal inputs. Important indicators for a human partner for example are changes in paralinguistic features, facial features or gesture measurements of the interaction partner. The practical implication is, that certain measurable physical values, such as speech frequency, pitch or volume are influenced through the inner state of the speaker [Kappas et al. 1991; Dellaert et al. 1996]. Similarly, visual cues such as movement of the hands or changes in the facial features fulfil a similar indicator in the visual space [Reed et al. 2020]. Contrary to the hidden inner user states, these indicators can be comparably easily gathered by sensor arrays, such as microphones or cameras. The complex task is then to generate a correlation between these extracted features and the user state, without requiring pre-existing full discrete knowledge about the specific causes for the expression of the feature. The employed solution for this in this thesis is the use of machine learning architectures, which create categorisation spaces, correlating between features and user states. This is done by letting the system learn on real world examples of both features and user states. This self-learning is preferable in such complex and feature interdependent problem spaces where a purely manually described rule-based approach would not be practically feasible. It additionally provides the ability for the system to be future proofed, as the system may simply add new examples, either improving the capabilities of the system or adding new possible results or situations to the system as a whole. Typical inner user states are often exemplified by emotions, such as happiness or sadness, but can also include a wide variety of other states, like physical or mental exhaustion level or dispositions, like interest or dissatisfaction. With these indicators a technical system is then later capable of reacting much more personal to the user than simply on their measurable action.

In the area of machine learning methods, there is a wide variety of possible approaches, such as statistical methods, architectures based on Artificial Neural Network (ANN), and the current state-of-the-art of Deep Neural Network (DNN). While some of these provide a better incentive for small datasets, others require comparably big example sets to even function satisfactory but provide better generalisation results over these great amounts of data. Alternatively, other architectures allow for the addition of examples over time without losing the trained information beforehand or are specifically adapted for certain tasks or data forms. While the current trend follows a continuously increasing size of datasets, this often foregoes an adaption towards the specific user and forces rather a good fit for a large set of different users. As these measured expressions are often very individualistic for each different person, this can lead to accumulating errors in classification results, especially during continuous use of the assistance systems.

### 1.2.3 Peerlike Decision

The last aspect examined is the technical solution for a decision and behaviour control mechanism. As mentioned before, the general solution to approximate a human-like behaviour is to employ typically rule-based mechanisms. These use interaction relevant features, similar to the ones used during human to human interactions, as an indicator for which rule to choose i.e. which action to perform. This is of course still constrained by the ability of the system designer to predict and pre-plan all possible situations. In case of a true assistance system, which is designed to apply to a wide variety of situations, ideally even some not encountered before, a more robust solution is needed, which is also capable of inferring certain actions without prior knowledge.

One of the main problems which arise in the current state-of-the-art, and which inhibits more natural human-like behaviour by the system, is the lack of proactivity from the technical side towards the user. Most technical solutions require the direct input of the human partner to activate functions or to perform even simple tasks. A peer-like design on the other hand would, similar to a human assistant, perform certain actions before a problem appears, or even try to function prophylactically and reduce the possible impact of yet occurring problems.

A fit for these requirements are the so called cognitive architectures, which are briefly summarised in Section 2.3.2. These architectures are designed based on typical neurological processes and reasoning, such as memory functions, pattern recognition and experimental potential solution searching. This would allow a system to react in an understandable manner, similar to a learning human. Furthermore these are also designed on a modular basis, which would allow an incremental inclusion of new information or applications.

Part of this change is on how the internal processing of information is done by the system, also influencing the way the user recognizes the behaviour of a technical system in turn. While the former aspect is mainly concentrated on the abilities and assistive functions of the system itself, and how their implementation can be improved during ongoing HMI the later aspect is important for the user to be satisfied with the usage of the system and not feeling like the interaction is exploitative, one sided or unsatisfactory. As such the behaviour control aspect of this last functional block is not only there to increase the learning and influence efforts of the assistive function but also reduces unwanted encroachment on the self-determination of the user. This is a continuous effort between two opposing objectives, which need to be weighted depending on the specific situation. This also requires a much greater appreciation of the technical system for the human condition which goes beyond the current typical framework of reactive behaviour.

## 1.3 Research Aims

In this thesis several research questions will be answered. While the main structure and design, which will be examined, is based on a technical or virtual assistant system, or alternatively an integrated human-machine environment, this is only the primary application possible from this research direction, which could easily be expanded to other HMI systems. The individual experiments and topics are closed in themselves and present directly implementable results. The following section will describe the partial aspects and the overarching connection to the primary research aims of each and follows the architecture as presented in Figure 1.1.

The first layer of the overarching structure is the so called awareness layer, which concerns itself with all the available real world data, which has to be transferred into a technical readable format for further processing and interpretation. While this can take a variety of forms, it is concentrated in this thesis on the necessary steps to feed into a functioning machine learning solution in the second layer. Beyond the scope of this thesis is the technical implementation of sensor systems and interaction interfaces, as this would be primarily a hardware solution.

The research performed in this stage of the thesis is instead concentrated on the number and type of extractable features. Features in this case describe the measurable information which can, through the use of sensor arrays, be extracted directly from the real world and the environment in which the system is implemented. With the type of experiments mainly performed on acoustic and speech processing tasks, this includes the type of information taken from microphones, such as frequency and loudness. Similar experiments are possible in the area of visual information, both as sources but also as internal data representation, which are included in a less prevalent form.

The second layer of the architecture is the designated understanding layer, which is functionally implemented to interpret the information from the first layer, specifically in the context of a human-machine interaction. This includes understanding the inner human state, which cannot be measured but needs to be inferred. In practice other information can also be inferred in this layer, such as environmental situations, or meaning from speech-to-text recordings. These specific adaptions are beyond the scope of this thesis, but can easily be implemented by current off-the-shelf solutions or by slight changes to the presented solutions here. The main tool applied and examined in this area is the use and improvement of machine learning solutions and their implementation for complex technical classification tasks.

The research produced for this aspect applies to a wide variety of specific applications, with the common usage being the mentioned acoustic based systems. The first aspect is the implementation of the currently relevant deep learning approach in different formats, with an examination of the requirements connected with its application. The use of a continuously learning system as alternative to a fully pre-trained system is also part of this examination. A continuation from this is the area of convolutional neural networks, specifically in the case of acoustic based classifications, together with a more general examination on how visual representation of acoustic information may influence the technical interpretability. The last aspect is the direct application of the presented methods in the area of human user state detection or interpretation as a non-measurable size for technical sensors.

The last layer, the decision and control stage, is finally there to examine how the presented tools and methodologies may be implemented by the system itself to provide a more human-like assistive or interaction function. This includes the general interaction and engagement control, as well as a change from a pre-designed rule-based system into a self-adapting and acting system based on cognitive architectures, as mentioned before in section 1.2.3.

The research here consists of a practical implementation for a more proactively engaging system which actively tries to resolve situations of uncertainty instead of ignoring it, as in contemporary systems, as well as a theoretical examination of different engagement levels possible between a purely passive computer interface and a true active artificial intelligence, with the presented system here falling below the requirement for a system to possess true understanding but approximating it adequately for a human interaction partner.

## 1.4 Structure of Thesis

This thesis is generally structured around the basic information pipeline described before, the different aspects of the pipeline will be reduced to their constituent parts and examined in greater detail through relevant experiments improving and advancing their respective areas. This chapter provided the general introduction and motivation for the following thesis.

In Chapters 2 and 3 the basic terms and used datasets will be described. As the proposed structure and architectures are based on current state-of-the-art machine learning systems and several aspects of data preparation, these will be described in detail in Chapter 2, specifically concerning their general abilities and their applicability in this work. Equally important for the design of most machine learning systems are the available datasets and their use for the training and learning of the systems. In Chapter 3 these used datasets are listed with short introductions and explanations of their special properties.

The first researched aspect, when following the information pipeline, is the concept of feature extraction and data pre-processing as part of the overarching technical information gathering and processing phase. This is examined in Chapters 4 and 5, with the first one primarily dealing with the problem of feature selection, such as using alternative modalities as features and the reduction of features to the most relevant ones for specific tasks. In Chapter 5 feature optimisation and processing is examined as integrated part of a complete architecture and when this is the preferable option to pre-processing of the available data.

Chapters 6 and 7 focus on the second aspect of the information pipeline, which concerns itself with classifying tasks and systems capable of generating self-learned solutions. Chapter 6 examines the use of deep learning architectures and their advantages and disadvantages, specifically in the context of tasks with different dataset sizes available. In Chapter 7 the classifying capabilities itself are part of the examination, with special concentration on the recognition of inner user states and dispositions, as they are a primary part of a user-centric assistance system.

The final aspect of the pipeline, the control and decision making layer is presented in Chapters 8 and 9. Chapter 8 observes the implementation of a more proactive design for user engagement, based on the available information level, as well as possible adaptions for a multimodal technical agent environment. Chapter 9 finally explains the novel peer-like decision level, employable in an advanced assistance system and examines both the general architecture and the possible implementation of a behaviour control to optimise continuous interaction.

The final Chapter of this thesis, 10, is the conclusion and outro of the work, combining the examined aspects into an encompassing structure and ending with a short outlook for further research and implementation possibilities.

Chapter 2

# Employed Methods and State of the Art

## Contents



In this thesis the focus lies on the implementation of an advanced assistant system. This system utilises a range of different tools and methods to improve capabilities compared to typical current models. The main use of state-of-the-art tools was in the area of machine learning applications. These were employed to solve a variety of complex tasks which otherwise would require a wide array of specialised pre-knowledge and highly complex rule-based programming, which can be solved easier through machine learning alternatives. Another area was the general approach of an assistant system and how the underlying requirements change with the type of assistance required. This also resulted in a different style of decision architectures employed. This chapter will present the general base of current state-of-the-art research at the time of the writing of this thesis, which will be employed for the following research tasks. In Section 2.1 the general data pipeline for machine learning and decision capable technical systems will be presented. This is followed by the examination of machine learning solutions and applications in Section 2.2. Finally, in Section 2.3 the general level of assistance and autonomy in an assistance system will be discussed, combined with an explanation of the area of cognitive architectures and their influence in natural decision making by technical systems.

## 2.1 Data Preparation

Data Preparation describes roughly the first step of the aforementioned Data Pipeline, which describes the way information has to take from the real world environment into the technical system. This needs to be done in such a way that it can be processed further by other applications, such as classifiers, predictors or similar functional units. The general steps follow the route of finding and recording the data, which then needs to be digitalised and further pre-processed again to improve their interpretability. This is an integral part for the technical interpretation, as most systems are otherwise overloaded by the amount and complexity of the extractable data from the physical world.

### 2.1.1 Data Preprocessing

The first step is the search for and the extraction of the data. Because of the high amount of possible data, this would impose problems on most machine learning systems. An important tool, which is employed to reduce the computational load on such architectures, is the use of pre-designed feature-sets to choose the most relevant and impactful real world measurements for the following processing steps. These feature-sets are often chosen based on the type of problem most likely to be encountered by the system. In case of my research, where most experiments were based around emotion and affect recognitions, and under the constraint of using audio recordings of speech and voice as indicators, this leads to the use of typical sets like Opensmile's emobase (emobase) or Opensmile's emolarge (emolarge), which consist of pre-selected acoustic features which are proven to work in comparable applications [Eyben et al. 2010]. For these two examples, this still would still include between 988 and 6.552 individual features, such that further optimisations can be necessary, as was done in my research.

### Valuation of Data

The basis for current machine learning applications is the interpretation of patterns which occur in natural data [Koza et al. 1996]. In this case most systems employ a general data pipeline which is comparable between all different kinds of systems, independent of the type of employed data and interpretations tasks. The employed data depends primarily on the available sensory appliances available, which are often employed to mirror human awareness as a basis, both because of the pre-existing knowledge of human emotion recognition, but also because of the resulting interpretability and comparability of the technical results. It can build the base for the recognition of a wide variety of technical states, as well as an important indicator of human states in inter-human communications [Luo et al. 2016]. This basic knowledge of known correlations between features and emotions is often easily transferable into human-machine interactions.

Another widely available mainstay of information is the use of acoustic data, as a mirror of the human sense of hearing. This acoustic information also provides a wide array of proven correlation between the acoustic features and the internal state of the users, especially on human internal states such as emotions, affects and similar expressions [Espinosa et al. 2017].

Further data, either based in the other human senses or even further technological data is also available for a plethora of machine learning applications, including the area of Human-Machine Interaction (HMI). An important example is the increasing usage of "wearables" such as smart watches or similar appliances with bio-monitoring abilities, allowing technical systems a good insight into the internal biological situations of their human counterpart during an interaction.

When employing data as described for correlation, ideally former knowledge of causation is ideal, but even the employment of machine learning in "black-box" applications is possible, where the basis for the observable correlation is not known [Koza et al. 1996]. Independently from the lacking underlying knowledge, a machine learning system primarily only needs a distinctively unique pattern of features, which is similar to examples from the dataset. Depending on the type and source of data, each recorded example will contain the necessary information overlapped by different levels of noise or distracting information coming from technical and natural sources, such as background chatter or machine operations. To assure a reproducible classification, this requires a wide base of examples to reduce the influence of the noise, as well as to reduce potential bias and erroneous correlations based on these potential biased examples.

A prudent way to ensure the applicability of the available data is the use of mathematical proofs to assure that features and training data are connected on some level. One example, which was also employed in Section 4.1, is the Pearson Correlation Coefficient (PCC) [Pearson 1901].

$$r(x_j, x_k) = \frac{cov(x_j, x_k)}{\sigma(x_j)\sigma(x_k)} = \frac{\sum_{i=1}^{m}(x_j^i - \mu(x_j))(x_k^i - \mu(x_k))}{\sqrt{\sum_{i=1}^{m}(x_j^i - \mu(x_j)^2)}\sqrt{\sum_{i=1}^{m}(x_k^i - \mu(x_k))^2}} \tag{2.1}$$

The PCC is calculated by comparing two features $x_j$ and $x_k$ over an amount of $m$ samples, where $\sigma$ is their standard deviation and $\mu$ is their mean value. With these values one can determine the probability that both features correlate in some capacity with each other. The result of $r$ can vary between 1 for a perfect probability of linear correlation, 0 for no probable correlation, and -1 for a perfect linear anticorrelation. A distinction needs to be made that this is not a causal proof that two values are connected, but it can indicate a high probability for further research and applicability in a self-learning system. Conversingly, if there exist a causal correlation, then the Pearson correlation coefficient measures it.

### Feature Extraction

The successful implementation of machine learning solutions not only requires the availability of sufficiently informative datasets but also their optimal presentation for the system to latch unto contained patterns [Masmoudi et al. 2021]. The first step for this is the transfer from real world occurrences into technical readable format. Most state-of-the-art sensor arrays work on an electrical transmission and are easily compatible with typical computer architectures, which are also employed as the base for machine learning architectures. An important aspect of this kind of transfer is the error from an analogue to a digital system, given the high resolution possible with most data formats and the already given aspect of noise in relation to that, it is oftentimes negligible for the later pattern detection.

With the data in a digital format the system often prefers the processing into features instead of most typical recording formats. While formats such as .mp3 or .wav are efficient for the reduction of necessary space on the memory, they are less expressive on how the original information was perceived. Features, especially when used for machine learning applications, present the data in an easily interpretable form [Blum & Langley 1997]. Typical examples for acoustic recordings are the emobase features [Eyben et al. 2010], representing a set of typical extractable values usable for a wide variety of speech and sound based classifications. The return to an easily playable sound file is often not possible from a pure feature state. The search for an optimal feature set is a research aspect by itself, as shown in Section 4.2, as it concerns the computational and processable requirements of the general system. Typical features, for example, contain wavelets, Linear Predictive Coding (LPC) and Mel-Frequency Cepstral Coefficient (MFCC) [Vlasenko et al. 2008].

Using the emobase feature set as an example, the full collection consists of 988 different features, which are taken from a wide variety of sound features. A Low-Level Descriptor (LLD) in this case can stand for features like Intensity, Loudness, coefficients for 12 mel-frequency cepstrums, the pitch and envelope for the fundamental frequency, the voicing probability, the eight line spectral frequencies and the zero-crossing rate. It furthermore contains the derivative over time for these values. These are supported by features taken from a full recording such as, minimum and maximum values, position indices, the range of the LLD's, their mean value, up to two linear regression coefficients, the linear and the quadratic error, standard deviation, skewness, kurtosis, the quartile and the inter-quartile ranges. [Schuller et al. 2009].

### 2.1.2 Data Preparation

Data Preparation is the step after deciding which data is to use. As most systems and tasks require data from a multitude of sources, it is necessary to normalise and process them to a better and more equal form of representation. As the general architecture of machine learning systems does not distinguish between different features based on their type of origin, it is often more effective to normalise, i.e. limit, the value of potential input features into a specific range. Other aspects falling under this topic are the necessary annotation of training data for machine learning or the augmentation of data after their recording.

#### Data Normalisation

An important aspect of real world recorded features is, that they most often have strongly different values and dimensions depending on the used sensors and methods. While these differences may contain important information, most machine learning applications are not able to process such strong variations within the same process and with the same precision. Practically, this leads to high values receiving a higher weighting, even when the base of the feature itself is different. To equalise the impact from different features on the same scale, it is often preferable to normalise the values before processing them further in the system [Masmoudi et al. 2021]. This is done by mapping all values on a scale from zero to one, which requires a pre-existing knowledge of the general minimum and maximum values which may possibly occur. The solution for this is to examine a sufficiently great example set to determine the most likely extent of values, with small derivations from this norm having only a small impact on the later training. Alternatively a more complex standardisation can be performed with the following method:

$$s_i^j = \frac{x_i^j - \mu(x_j)}{\sigma(x_j)} \tag{2.2}$$

With this equation, where $x^i$ are the individual values, $\mu$ their mean values and $\sigma$ their standard deviation taken from the whole dataset $j$, a standardised value $s$ can be calculated, which instead of limiting the possible values, uses the deviations from a measured mean as an input for the system.

### Data Annotation

While the basis for all machine learning applications is the aforementioned ability to detect patterns inside of data, creating correlations between information and generating a conclusion on these points, this does not imply any kind of understanding from the side of the machine. Specifically the “black-box” nature of such systems, resulting in the lack of comprehension for the generated results and the inability to determine the correctness of the created solutions is an often mentioned disadvantage. To assure that a system correctly determines its results, especially in areas where there is no easily definable ground truth by only technical means, human experts have to fill in the necessary information [Russo et al. 2021].

The most apparent aspect of this is the initial annotation and preparation of the data. As mentioned before the data is generally transformed into features which are often based on human perception and understanding of the underlying, human, classification process. This ideally leads to a technical process which mirrors the human mental processes to a high degree, allowing the observer to measure the correctness of the approach on this merit.

On top of that, the process of annotation itself is necessary for all systems, as it allows the classification of topics such as emotions, mental loads or affects which are not based on physical but on psychological grounds and as such are not directly measurable. In this case every example has to be declared as belonging to a specific class or a set of classes. As this is directly a result of the decision of a human expert, this can easily lead to biases and skewing of the dataset [Pandey et al. 2019]. This effect is increased further when a great amount of differing individual experts work on different parts of the same datasets, something which is often necessary for datasets of greater size as it is required for more complex architectures.

Different methods can be employed to reduce this bias, such as consensus of experts with overlapping data points or internal control and rating of the expert panel to measure the confidence and correctness of the given annotations. While this does not remove the possibility of bias, the aggregated methods reduce the inner contradictions of the dataset, which improves the training possibilities.

Another method to ensure the annotation of large datasets and the involvement of only few annotators is by employing automatic or semi-automatic annotations. For this an inertial classifier sorts the examples pre-emptively in groups, as well as giving a score of confidence while the human experts only declare the correct name of any group and judges the examples with a low confidence. Such methods are especially useful for current big-data solutions, which otherwise are difficult or impossible to process with a sensible effort.

### Data Augmentation

The training and continuous improvements of the classification results are primarily dependent on the quality and quantity of the available examples. With exemplary recordings, such as the ones presented in Chapter 3, this is often done on a certain subset of examples or sources, such as specific speakers. These are ideally taken from a variety of backgrounds to ensure that the generated datasets and the resulting pattern based classifier are prepared for all the possible occurrences when the system is employed in a real environment.

In case of acoustic speaker information, the expression changes based on biological and psychological backgrounds of the speaker significantly. Influences range from age, sex, weight to conscious and subconscious behaviour [Gross et al. 1997; Pisanski et al. 2016]. All this leads to variances in the extractable features, which then provide the basis for the latter classification. Faulty or too small observations in turn lead to falsely trained patterns.

As no training set can encompass all possible varieties of occurrence, the extracted examples often get further modified. Practically, by adding noise or variations to the original examples further training examples get generated. The important aspect of this is the realistic base which ensures the similarity with other non-recorded examples. This method is called data augmentation and is especially important for tasks where data is difficult to acquire or for complex machine learning architectures which require a great amount of examples to solve the problems of latching and overfitting [Janiesch et al. 2021].

An even more advanced version of data augmentation employs the pattern finding capabilities of the architectures themselves to create new examples which are more abundant and at the same time more realistic than the simple addition of noise or the variation of the original values. These are the so called Generative Adversarial Network (GAN) [Goodfellow et al. 2020]. These employ architectures like the Convolutional Neural Network (CNN) from Section 2.2.1, in a twofold way by creating and distinguishing the data at the same time. The creator improves its abilities congruent to the classifier. This method still requires original data to ensure the authenticity of the generated sets, but otherwise provides imitations which are also hard for human experts to distinguish.

### Krusckal Wallis

A method employed to measure if two sets of features belong to a different occurrence is the Krusckal-Wallis Test. Specifically it gives a measurable rating if samples are taken from the same distribution, through an applied one-way analysis of variance. For this it applies a null hypothesis that all samples are from an equal set, while the significance on how strong this hypothesis is disproven is then a value for the difference. The advantage of this method compared to similar approaches is that it potentially allows for more than two sample sets, and that the original set does not need to be a normal distribution for the method to work.

$$H = \frac{12}{N(N+1)} \sum_{i=1}^{g} n_i \bar{r}_i^2 - 3(N+1) \tag{2.3}$$

In here, $N$ is the number of all observations, $g$ is the number of groups, $n_i$ is the number of observations in the specific group and $\bar{r}_i$ is the average rank for all observations in this group. With the results from this equation it can then be compared to the general $H$ for the whole distribution, should the $H$ for the subsample be higher, the null hypothesis is disproven.

## 2.2 Applied Machine Learning

Machine Learning is an area with a wide range of applications. It also represents a state of the art approach to solve complex classification and interpretation tasks without requiring a full and in-depth understanding of all possible causations in a topic. Both the used architectures itself and their application are continuously expanding in their capabilities. This section examines first the different architectures used in this thesis, followed by the possible applications.

### 2.2.1 Machine Learning Methods

The machine learning methods employed in this thesis present only a partial representation of the full available set of possibilities. They are presented roughly in order of complexity and advancement. Because of their different priorities in their design they allow for different applications to a variable degree. Important for the correct choice of machine learning applications for different kinds of classifications tasks are the specific situations around it. Depending on the amount and type of available data, certain architectures are more or less capable of processing information. Specifically the complexity of a feature space, the multi-dimensional space build from the extracted feature values, needs to be separated during any classification task. Less complex tasks which use less features and can be separated linearly between classes also require less computational processing power, while other problems require the use of kernels to transfer the feature space from one representation to another. A connected problem of more complex architectures is the added requirement of greater data size to fill the feature space with examples. The necessity for the system or its designer is to choose the optimal options from the available options.

The ability of a system to provide the best solution itself depends on the chosen evaluation method, as can be seen in the Table 2.1 there is a wide variety of methods [Powers 2011]. Depending on the task, precision or recall can be more important, especially in cases where false positive classifications may lead to further problems or increased risks. As most datasets are not equally distributed in their composition, an unweighted or weighted approach may be necessary to account for this. The weighting of the measures is necessary when the distribution of the tested samples is skewed towards specific classes, as otherwise the resulting positive and negative cases are also skewed.

**Table 2.1:** The ability of a system to provide the best solution itself depends on the chosen evaluation method, as can be seen there is a variety of methods. Depending on the task, precision or recall can be more important as either false positives or false negatives impact the evaluation more.

| Name | Equation |
|---|---|
| Recall | $R = \frac{TruePositive}{TruePositive+FalseNegative}$ |
| Precision | $P = \frac{TruePositive}{TruePositive+FalsePositive}$ |
| F1 | $F_1 = 2\frac{P*R}{P+R}$ |
| Unweighted Average Recall | $UAR_n = \frac{\sum_{i=1}^{n} R_i}{n}$ |
| Unweighted Average Precision | $UAP_n = \frac{\sum_{i=1}^{n} P_i}{n}$ |
| Unweighted Average F-Measure | $UAF_n = \frac{\sum_{i=1}^{n} F_i}{n}$ |

### Statistical Methods

The first topical group of machine learning architectures works on the statistical approach [Lee et al. 2002]. As basis for the structure pre-determined expert knowledge can be used, but the currently more topical training with examples and supervision is also easily implementable. Besides other methods such as Bayesian networks [Pearl 1985] which work on probabilistic solution where each result is assigned a probable connection to each possible class, other statistical methods such as the Hidden Markov Model (HMM) [Huang et al. 1990], assign only the most likely result. The method with the highest relevance in my work is the Support Vector Machine (SVM), which is one of the traditional machine learning architectures [Cortes & Vapnik 1995].

This method works by separating the original feature, or problem, space into separate areas of classes. The so called support vectors in this methodology “support” the partitioning faces between these areas and are refined through training steps. Every example the system receives allows the SVM to further move the support vector, in an effort to maximise the distance between each example of different class and the partition face.

Further refinements of the process also allow for the use of kernels, used to change the representation of the problem space into a different arrangement, which is necessary to allow for non–linear separations, which are relevant for many complex problems but are not possible with the original (linear) SVM architecture by itself [Shadeed et al. 2018].

Because of their general capabilities and the wide variety of possible applications, a SVM is often employed as baseline application and source for comparison results when using other machine learning solutions [Schuller et al. 2009]. As seen for a variety of different applications, it often shows above average results while also being comparably undemanding on the required computational power compared to most current designs of deep learning architectures. Importantly, it also is easily scalable to different data sizes and feature numbers.

An important aspect of this architecture is the relatively easy implementable ability to continuously adapt the generated result, as used in Section 5.3, for this the generated separator slowly changes its location and orientation, in the same way as during the original training stage [Xing et al. 2015]. This can generally work in two ways, either by remembering the original examples and adding the new examples, ensuring both a continuously high classification rate, but also an increasing storage and processing requirement, alternatively only the last examples will be applied. With this approach the training time remains constant, but it may increase the potential bias contained in the new examples by "forgetting" the original information when both sets diverge.

An alternative to these methods is the use of Random Forest (RF) as a classifier system [Breiman 2001]. With these a decision task is separated into several accumulative decision steps leading through the feature space. While this method proves adequate for most tasks it has fewer options for optimisations concerning specific datasets.

While both methods prove quite capable of solving classification tasks, both have their weaknesses especially in higher complexity tasks. RF for example tend to overfit on training data. With kernels SVM are capable of providing classification results even in complex feature spaces, the computational requirements tend to increase exponentially due to their complexity.

### Neural Nets

The most frequently employed methods in this work are based on the Artificial Neural Network (ANN) [McCulloch & Pitts 1943]. In contrast to the formerly mentioned methods ANNs are not based on a purely statistical approach but on a technological interpretation of the biological neural structure. Not only does this possess a unique insight into the underlying neurological processes as happening in the human brain, it also allows for much more variety and options in the area of learning and training of patterns and correlations. The underlying structural unit is a singular neuron, consisting of input, core and output, with the ability to change weight of the input connections and the activation inside the core can be adjusted. This relatively simple processes allows for variable levels of complexity by connecting and layering the singular neurons into full architectures.

The relevant aspect of all artificial neural networks is their depth and their width, as the processing and pattern recognition capabilities are directly related to these values. A singular neuron is practically capable of separating a feature space linearly into two, interestingly capable of mirroring simple logic gates [Yellamraju 2013]. As a rough rule each further layer allows for greater complexity of the separation, while the width of each layer allows more parallel processing pathways to generate. As such the simplest structure of an artificial neural network consist of an input depending on the number of available features, an output depending on the number of classes while the hidden layers in between can be adjusted depending on the general complexity of the problem at hand.

The training of such networks is based on the principle of supervised or unsupervised learning. The most relevant method in this work is the supervised approach, in which the system receives correctly annotated examples of the observed problem. Based on the perceived error between the correct network results and the ones currently produced by the architecture, the backpropagation allows to assign the influence of each weight in the network on the final error of the system. Based on this the system can then change this weight to close the gap with the projected result [Li et al. 2012].

This mathematical approach is also the base for the problem of overfitting which occurs when the generated process is only applicable for the applied examples, while ideally the generated process should be applicable to all similar occurrences.

The alternative approach of unsupervised training is applicable for a problem which does not have an annotation and instead reverses the process. Similarities and patterns get grouped into the same class, which is also called clustering [Du 2010]. The underlying formula for all neurons is the simple equation:

$$net_j = \sum_{i=1}^{n} x_i w_{ij} \tag{2.4}$$

where $net$ is the sum of all the inputs $x_i$ multiplied by their trained weights $w_{ij}$. This training process is controlled by the learning rate $\alpha$ and the error between the correct solution of the network $y_i$ and the currently calculated output $o_i$ over all examples $n$ given.

$$w_i = w_i - \alpha \times \frac{\partial E}{\partial w_i} E = \frac{1}{2}\sum_{i=1}^{n}(y_i - o_i)^2 \tag{2.5}$$

In contrast to other classification methods the computational complexity in ANN is relatively low as each neuron only performs one multiplication over all their input values. Important in this examination is that the amount of neurons necessary to classify complex task increases continuously, and the comparably more complex training phase. As backpropagation requires error calculation back through the whole network to the input values, training takes exceedingly more time than the forward passes.

### Recurrent Methods

A problem connected with the described typical neural network is the static definition of the input size before the system is initiated. Based on the way patterns are processed, a system would ideally always receive the value from a specific feature on a specific input. This is practically impossible when processing input of a variable length, such as word recordings or similar dynamic occurrences. A technical similar problem, even when not immediately perceivable, is the ability of a system to remember previous occurrences or dynamic changes during a single recording. With a static input size adjacent time steps may be ordered in a certain static way, but for a dynamic method the system itself would need to remember the change happening at the moment and how it may influence the final classification. As a result the development of Recurrent Neural Network (RNN) was done to solve this problem of a remembering system, or more precisely a system capable of "recurring" former information over several time steps [Elman 1990; Chen & Chaudhari 2009].

The important part of this method is that the output of certain neurons is fed back into them, alternatively the input of the full network, for each time step. This allows not only the remembering of the information of the input itself, but of already processed information which has less space requirements in comparison. When training such a system, the backpropagation method not only retraces the error quotient through all layers, also called trough space, but also through all the time steps which have happened before, this is called through time. The result of this method is a continuously smaller input of the former data on the result of the network after each iteration, as the influence of the former information loses its value over time. Specifically, with this architecture the result is not the same as an input layer where the full information is applied at one time, but the time sequence itself influences the impact of information on the system.

An important disadvantage of this method, compared to the preselected input width, is the increased complexity, when the normal training is done for a weight, the passed-on errors require the calculation to be done for each weight times each time step [Williams & Zipser 1995]. This means a further increase with any further duration the information is retained inside the system. This is not the only drawback, as the mathematical recurrent factor reduces itself to an insignificant part which is practically ignored by the system.

Regarding the computational complexity for the forward calculation, this is similar to the regular architecture, as only the last result is added to the input values. The training in contrast is much more complex as the errors are not only calculated back to the input but also through former time steps for each input which was given the system before.

### Long Short Term Memory

As shown with the recurrent neural networks, remembering with ANN is possible but poses a variety of problem both in capabilities and performance wise. An alternative, which is more recent, is the implementation of LSTM and their variants [Hochreiter & Schmidhuber 1997; Wöllmer et al. 2009]. This architecture employs normal artificial neurons in a specific arrangement mirroring in its function a memory cell. Instead of using recurrent connections, the memory is concentrated in a so called core which returns its information indefinitely inside, instead of the continuously vanishing gradient which otherwise occurs after several time steps. To control the content and achieve better specific adaptions for a task, the cell is controlled by so called gates which allow the outside of the memory to either write, delete or read from this cell, these gates are also artificial neurons and trainable, the general architecture can be seen in Figure 2.1. An important distinction is that the error from former inputs is not propagated through time but only till it reaches this memory core, as such it is only propagated through space.

Not only does this architecture achieve better results for a variety of applications compared to the recurrent architecture, it also is much less complex in its computational requirements, especially for larger architectures. As mentioned

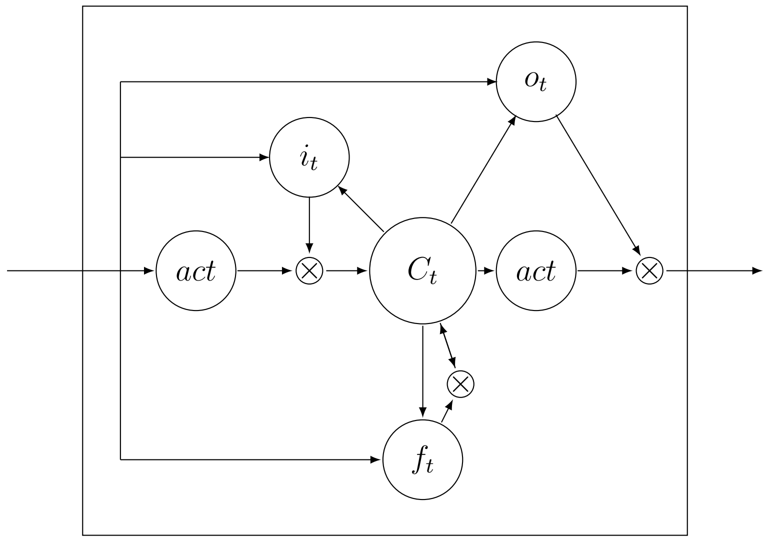



**Figure 2.1:** A Long-Short Term Memory (LSTM) module, the different parts are often regular neurons. The gates are as follows, *act* is the activation gate interpreting all incoming values, $i$ is the input gate regulating the influx of information into the memory core, $C$ and $f$ are the core and the forget gate respectively which allows the system to reset. The output is then controlled by a final activation neuron and output gate. All of the gates access the whole input features.

each required time step increases the load exponentially in recurrent architectures, while the increase in complexity for LSTM architectures remains linearly for each increase in functional units. As a singular LSTM cell consists of much more neurons than a recurrent unit this advantage increases with the size of the network and the amount of time steps which are processed.

Based on the general idea of the LSTM architecture, a wide variety of variants and improvements have been developed. Specifically the complexity of a singular cell, consisting of core, gates, in- and output is comparable high, both in computation and implementation. To solve this, simpler alternatives which aggregate gates over several cells or less complex activation functions are in use. These methods often retain the capabilities of the original structure while requiring less computational power.

### Convolutional Neural Networks

Based on the general trend towards high data sizes and increased complexity in the processed tasks, typical standard neural networks are often not capable enough. Instead with the rise of big data applications the current state of the art shifted to the so called Convolutional Neural Network (CNN) which provides a much more complex approach, both in the applicable level of the problem but also in the computational requirements during the processing of information. The most important aspect of this architecture is the method of convolution, which describes a form of processing areas of input into a different more efficient representation [Krizhevsky et al. 2017]. The primary form of input is per design an image or image-like representation, partially based on the biological model it was based on, the visual cortex of mammalians. In turn the convolution is especially useful to process information in 2- or even 3-dimensional formats. The functional process of the convolution uses a moving kernel, similar to a filter, over the input and generates, also similar to a filter, a correlation or pattern from this kernel. In case of visual input this can lead to effects similar to edge detectors or comparable visual filters, with the benefit that this is self-learned by the system [Krizhevsky et al. 2017].

When using large and complex datasets, CNN are potentially able to perform self-trained pre-processing steps which otherwise would need to be done manually by human experts. An interesting aspect of this convolution is the effect, that while the size of the feature field reduces with each convolutional layer, the depth increases significantly, as indicated in Figure 2.2. As each kernel produces a different convolutional result per layer, this increases the amount of informative values per layer roughly at the same rate as the number of applied kernels.

To reduce the computational requirements of the system, the preferred activation function for CNNs is the Rectified Linear Unit (ReLU) which is easier to compute, and while also lacking certain effects in comparison to the usual ones, tends to generate better results in the typical application fields of CNN [Agarap 2018].

Another effect typical for CNN architectures is the frequently occurring overfitting, as the learning capabilities, specifically the retaining of patterns, is very high it often leads to the perfect classification of all training examples as they are practically saved as trained in the system. To avoid these effects, greater amounts of training data are required. In case of enough examples, it leads to the currently top achiever of robustness and accuracy [Rahman et al. 2019]. Alternatives when using smaller sets require the implementation of effective data augmentation methods, as well as drop-out layers, which allow the system to develop into a more robust classifier.

A disadvantage of the method concerning its high classification results is its general low retraceability during the creation of the classification result. A widely known example of this is the case where a system was supposed to distinguish between housedogs and wild wolves [Nguyen et al. 2020]. While the system provided good results, a later analysis showed that the main distinguishing factor was the background consisting of either both woodland and snow or house gardens. Without closer analyses of the training results such errors are hard to detect.

One possibility to solve this problem is the method of deep dreaming [Spratt 2018]. This method describes the possibility to rerun the training process in reverse where the generated convolutions picture the extracted patterns onto an image. With this method the training process can be made visible and certain errors during the training process can be solved before they impact the finalised system. Compared with other ANN based systems, the CNN is of similar

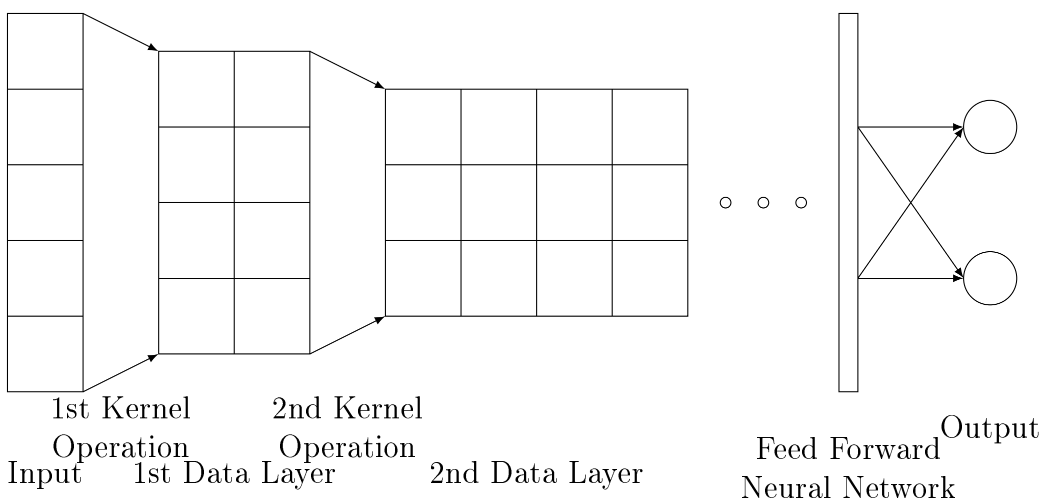



**Figure 2.2:** A typical CNN network, an important distinction from the other networks is the use of “depth” in the hidden layers. With each new layer of the network the architecture allows for further reaching patterns in the original input values to emerge.

computational complexity. Its separate neurons still only process a simple multiplication, of input values, but in case of the specific architecture the amount of neurons increases exponentially with each layer and the number of kernels applied in each of these layers.

### 2.2.2 Application of Machine Learning

The reasons for the usage of machine learning solutions in the recognition of human emotion are plentiful. Primarily the self-training capabilities of a ANN allow for even correct classifications when the underlying correlations are either not known in detail or when their description for a truly rule-based approach would be too complicated to implement. Human emotions as such are difficult to declare easily because of their great interdependence of the available features, as well as the high diversity between the expressions of different humans. Additionally, it is an aspect which is fully strange to a technical system as it depends more on psychological expressiveness than purely logical causations. As such emotions by themselves are not directly measurable, they have to be decided based on alternate feature values per correlation. Indicators for such expression can be loudness, word frequency or similar speech variations.

#### Emotional Classification

As there is no objective measurement of emotion the classification itself has to be decided based on subjective criteria. Using the psychological area of research as groundwork, several different description models can be declared, which concentrate on different objectives [Scherer 2005; Hoffmann et al. 2012; Moors et al. 2013]. The relatively easiest method assigns different names to a specific range of expression, such as general happiness or sadness, but also fear and boredom can then be declared as an emotional group. The advantage of the clear definition is the ease to interpret the result, with a happy user distinctively different from a sad one. The main disadvantage is the high subjectivity, both of the annotating experts but also during the decision which expression constitutes an emotion, or vice versa which emotion contains which expression. As such the number of discrete emotional classes range from the typical “basic emotion” seven class grouping, which is often employed as baseline [Davidson 1994; Ekman 2005; Schuller et al. 2009], to more detailed representations. These are often happy, angry, anxious, fearful, bored, disgusted as well as neutral, with variations of singular emotions in some case. This is but one possible separation among many for the distinction of emotions, with the advantage that it allows for a wide variety of distinctions without going excessively into the psychology of human emotions, which may prohibit otherwise an easy technical classification [Plutchik 2001]. Another approach using even less classes is the valence-arousal axis which reduces emotions on either high or low expression of valence and arousal [Mauss

& Robinson 2009]. Theoretically, all possible distinct emotion classes could be located on the space which is created by the two axes. Corresponding diagrams have been proposed, amongst others, as Circumplex models [Russell 1980] and, with additional strength dimension, as Geneva emotion wheel [Scherer 2005]. This allows for both better training prospects, as well as a more granular result. This in turn is also a disadvantage for the annotator who has to be capable to objectively distinguish valence and arousal from audio recordings.

#### Affect Classification

While emotions are not only a significant part of paralingual interactions, they are as well highly indicative of the state and the reaction to any given situation. For this they are not the only aspect which would be useful for a technical system to extract during an interaction. Parallel, and partially overlapping, to the idea of pure emotion recognition is the wider aspect of affect recognition [Picard 2003]. While this usually includes user states or affects such as happiness, or similar emotional classes, it additionally includes a variety of other user states and expressions. For example tiredness or attentiveness are expressions of affect but commonly not included in traditional emotion recognition. The psychological definition for affect includes, besides emotion, the general feeling or the mood of a person [Davidson 1994]. Especially when designing interactive agents such knowledge about the user affect would be highly important, which is the reason they are included here.

For interaction control, an important aspect which follows from this is affect recognition. This is the ability to discern interest and general trend of an interaction from the user, practically employing cues to predict further developments before it happens. In conjunction with more advanced personal profile projections generated by observing former interactions by the user, this allows for systems which can predict whole interactions and interests, optimising the reaction times and the accuracy of the provided assistance.

## 2.3 Human Machine Integration

Given the former aspects of the user state identifications and user intent interpretation, it is a logical conclusion to possibly provide specific assistance based on this information. The general idea combining the requirements of a human user and the capabilities of a technical system are even older, practically beginning with the idea of a personal computer. In this section we examine the increasing trend for better technical assistance and the possibility to provide a human-like technical system.

An important distinction to be made is that the following examinations focus on one-to-one interactions between one user and one technical system, specifically multiparty scenarios with multiple agents (human or technical) are not part of this or only mentioned as potential addition. It is furthermore not specified in which form the user interface of the system has to be designed, with potential scenarios ranging from simulated (e.g. on a screen) to embedded (e.g. as ubiquitous supports with vanishing interfaces) implementations.

### 2.3.1 Assistance Systems

The general idea of assistants' systems changed during the ongoing development, both in its capabilities and requirements. One of the original examples was the simple technical assistant in the form of handheld devices or as applications of personal computers which provided simple assistance functions. These first systems provided similar functions as usual calendar, notepads or similar text databases. The assistance effect is achieved by the ease of transfer of the data and the transportability of the system itself. With these the databases access is purely done through the manual activation by the user. Compared to later approaches, the assistive capabilities are rather low, as the system itself is not providing any supporting work beyond the easy collection of different applications and the fast access to the saved data. Simple additions allow the further possibility of alarm functions or similar pre-programmed reactivity on the inputted information.

This development was parallel to the idea of HMI systems itself [Karray et al. 2008], as these changed over time to provide better and easier access than other manual methods. In this first stage the main focus was the improvement of interface technologies, going from keyboards to ubiquitous implementation of voice control [Preece et al. 2015].

#### Personal Assistant

Built upon this first idea of assistant systems is the current approach for technical or virtual assistant systems. This also presents the main form of commercially available products, such as Amazon Alexa or Microsoft Siri. The improvements compared to the former approach are significant in their architecture and the resulting capabilities of the system, while the underlying requirement what an assistant system is remains the same. One of the main improvements is the area of interaction, which is particularly far developed for voice control and interaction [Dekate et al. 2016]. The idea of natural interactions, similar to the ones between two human agents, is the main reason for this development. Further additions to this interaction process allow for mimic or gesture control, as well as the use of gaze. With more integrated environment and the current increase in wearable technology bio-signals themselves become a viable avenue of control.

The assistive capabilities of the system remain in contrast close to the former abilities, with a further ease of database access and control operations. The main functions in a mobile application are still the access of database information, either external such as lexical knowledge, or internal in the form of e.g. calendar information. The observable trend towards assistance controlled applications follows these general capabilities, where most applications either provide access to specialised databases or provide a specific function for the personal information. This includes the access of radio stations or music streaming, as well as the control of alarm functions based on time and date [Lopatovska et al. 2018].

A development based on the current stronger integration of these assistance systems in the stationary system, as well as the combined development of technical home appliances which leads to the idea of a integrated smart home [Elsholz et al. 2009; Kameas et al. 2009]. Smart home system, additionally to the mentioned aspects, allows the user also to control the appliances in the household, either by the ease of access given by the voice control or by pre-determined date and time control. This is often designed in such a way, that the individual appliances are all controlled trough one central system, which is easier to access and implement than several systems working in parallel to each other. This also allows for an easier communication between the different systems in case of interdependency such as temperature dependent heating units or the control of lamps depended on the remaining daylight.

An important aspect of this current development is the increased integration of the systems into the lifestyle of the users, which by design are invited to depend and trust the system with information and control. This mirrors the general request for true assistant systems, which are much more integrated into the life of their users.

### Companion System

The idealised approach to an assistant system goes far beyond the necessity of a voice controlled database access. This approach particularly is also called a companion, as in a continuous coexisting assistant and supporter besides the user [Wilks 2005; Biundo & Wendemuth 2017]. Alternative descriptions can be of a caretaker or an overseer of the users actions and life [Merten et al. 2012; Dojchinovski et al. 2019]. Compared with the current usual approach it requires a full separation between the user control and the system control, while there is an overlap, specifically when the user describes their own priorities and objectives to the system in detail. The solution and processing of these tasks is dependent on the capabilities of the system itself, as it has to proceed based on its range of actions to solve these tasks. Alternatively, the system may continuously follow the actions of the user and may engage to help and support in opportune times, specifically when the user is in distress or undecided on how to proceed.

The idea of a companion as such goes beyond the aspect of controlled assistance systems, as these only provide an easy and fast access, while the companion is a true attendant to occurring needs during the lifestyle of the user. While this may appear either as a silent observer, when no assistance is needed, or as an active engaging support, when the user is approaching a problem, or when the user and the system cooperatively engage a task on their own.

Generally, the level and approach of the assistance is changed from a reactive to a semi-active paradigm, in this the activation is still primarily dependent on the user input, but the decisions' finding and the provided support is actively chosen and provided based on the current capabilities of the system and the most likely necessary requirements of the user based on in-depth user profiling. Additionally, such a system is generally already described as a virtual agent [Traum et al. 2012] as it provides a form of actor in the human-machine environment.

### Peer System

The last level of assistance system lies even beyond a companion [Weißkirchen et al. 2020a], which requires generally a more independent approach. While the companion prioritises the immediacy of continuous support, and retains the general improvements of easy access, through voice or similar interfaces, it is still highly dependent on the user instructions. The peer level of approach instead puts a system not only on a similar but an equal level to the user [Weißkirchen et al. 2020a]. This is also part of the development into the active approach of assistance tasks. This is only possible through the implementation of system internal objectives and priorities instead of relying on the user instructions. An additional aspect is the change for a system to approach understanding of certain situations, requiring the use of cognitive architectures instead of the rule-based solutions in purely reactive systems [Lew et al. 2007].

Without the instruction by the user, the responsibilities of the system have to be pre-decided, before engaging the system, to assure the safe usage of such a system [Weißkirchen et al. 2020a]. As it is not required to query its user before engaging an action, it also retains all responsibility for potential errors. This allows for actions before the user is aware of the potential problem, but may at the same time influence and impair the user with problems which otherwise would not have occurred. Depending on the accuracy of the system it may be searching problems, and decides to have found them, when this is not the case. In such a situation the system may impose its own biases onto the user.

The base for such a system is a truly independent agent, which would potentially proceed even without user interface in a closed environment. In its architecture as an assistance system this takes the form of several baseline objectives to assist, while the precise implementation of these assistive functions would be decided during runtime. This kind of indirect planning is directly connected with the idea of cognitive architectures which no longer require a specific task but instead mirror a human decision process.

The only remaining level of improvement would be a true Artificial intelligence in the strong sense, which would decide on its own free will to support a human peer. This is far beyond the scope of this work and the current technical level.

### 2.3.2 Cognitive Architecture

The solution for a more interactive system is to change the decision making process of the architectures from a rule-based approach to one more closely mirroring the human way of thinking. While the learning capabilities are already close to human abilities and directly influenced by them, most systems controls depend on the typical if-then causal programming. Most available applications and services employed in assistance systems also employ this typical paradigm for processing information and control database access. Contrary to this approach human behaviour is normally more faceted and based on wide variety of outside influences, prior experiences and explorative randomisation. To simulate this kind of behaviour different architectures are required. The examples which are directly based on the human cognitive process are mostly under the overarching topic of (technical) cognitive architectures. Similar to the fuzzy logic of machine learning applications, cognitive architectures also employ probabilities and relation between situations and reactions [Anderson 2007].

The advantage of the human approach is that such a system does not need to be fully programmed for all situations before the implementation. Instead such a system is capable of generating new solutions based on the knowledge of prior possible solutions applied to the current scenario [Anderson 2007]. This allows continuous work under uncertainty, approaching similar or new problems during the run-time and practically applying the learning effect to internal decisions. Several different architectures exist which employ this kind of process. For ex-

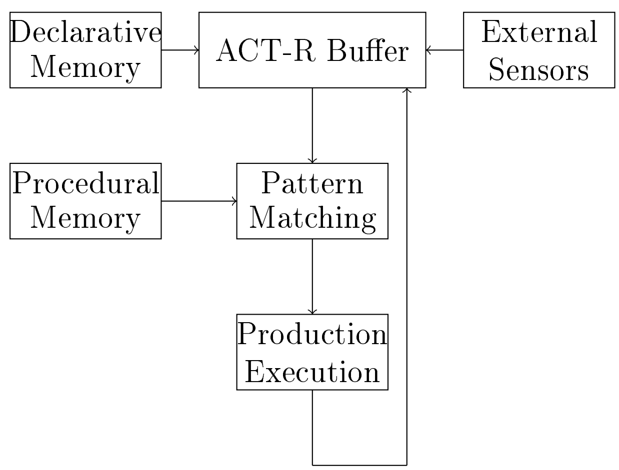



**Figure 2.3:** A variant of the modules and buffers of the Adaptive Control of Thought—Rational (ACT-R) architecture, similar to the requirement for the projected assistance system. Figure based of [Anderson 2007].

ample State, Operator Apply Result (SOAR) [Laird 2012] or Adaptive Control of Thought—Rational (ACT-R) [Anderson 2007], but also BDI [Rao & Georgeff 1991] architectures fall under this moniker. As ACT-R is one of the examined systems in this work, its architecture can be seen in greater detail in Figure 2.3. The specific design of ACT-R employs an array of different memories containing all the information about the situation and the applicable solutions, as well as a situationally dependent task selector. The range of the system goes from a purely database based system to a plethora of available interfaces into the real world, such as sensor and motor functions. While this kind of system is primarily employed for psychological comparison studies and virtual assisted teaching assignments, the underlying principle is applicable for most typical interactive systems [Anderson 2007].

## 2.4 Summary of the Chapter

In this chapter the general methods and topics were examined and described, as well as the state-of-the-art during the writing of this thesis was established. Specifically the methods concerning the data preparation and machine learning architectures were presented, as they build the basis for later research without being a direct result from my research, even when certain adaptions of these mentioned methods were done as shown in the later chapters. Beside the mentioned processes and tools, other methods were employed which did not warrant inclusion in this chapter as they consist of typical tools of the trade. Before the implementation and the research into this topics will be concluded in Chapters 4 to 9, the used datasets and databases will be presented in greater detail in the next chapter first. The research itself is similarly ordered to the presentation in this chapter for easier comparison, both based on the general information pipeline found in the presented assistance architecture.

Chapter 3

# Datasets/Setups

## Contents



In this chapter the used datasets for this thesis will be presented and shortly examined. As most experiments were done on acoustic emotion and user state detections, this is reflected in the chosen datasets employed. Given the nature of machine learning solutions, the chosen datasets also significantly impacted the possible results and approaches which were chosen. A dataset, as used during my research, normally consists of audio file recordings taken from different speakers, as well as annotation tables describing the contents of the recording. Together this can be used as a ground truth for training the used machine learning implementations.

All used sets can roughly be separated into two groups. The first group will be described as acted data and is examined in Section 3.1. This group consists of emotions and user states which are consciously performed, or acted, by the participants, and as such are very expressive and idealised in their form. The other group contains the naturalistic datasets, and is examined in Section 3.2. These consist of data which was either recorded during real world situations, prohibiting a conscious acting on side of the participant, or by inducing certain user states during an experiment, and as such more natural. These are more complex in their generation, but resemble more closely the type of data occurring in real world applications than the acted ones. As a result they are often also more complex to classify, due to their realistic properties [Vogt & André 2005]. This is followed by a summary of the chapter in Section 3.3.

## 3.1 Acted Datasets

Acted datasets often provide a variety of advantages over the naturalistic kind. Foremost it allows for greater control over the recorded state, as well as a predetermined annotation in the form of an acting script. Distribution of examples can be easier equalised for each provided class, allowing generally for a more robust and easier training approach. As acted datasets are also often recorded in prepared environments, and in some cases by trained actors, they also easily provide a very high quality of recordings. Practically, this translates into less background noise and normalised recording parameters between the speakers, such as similar volume, echo and etc. for all recordings. The provided examples for these kind of datasets consist of the Berlin Emotional Speech Database (EmoDB) and eNTERFACE'05 (eNTERFACE) corpora, as well as the Speech Under Simulated and Actual Stress (SUSAS) corpus. Both of the first two are examples of datasets with emotion classes and are also good examples for datasets utilized in benchmark comparison experiments. The latter is an example of a recording on how stress affects the voice of a speaker.

### 3.1.1 Berlin Emotional Speech Database

One of the most frequently used datasets in my work is the one called EmoDB [Burkhardt et al. 2005]. It is a typical example of an acted, non-induced emotional database, and based on its clarity, consistency and age is often used as a baseline comparison set for different architectures and methods. A speciality of this data is the original reason for the creation of it, which was the basis for a speech synthesis method with contained emotional paralinguistic aspects. In its use as a baseline dataset it is employed in a variety of benchmark research for machine learning applications and similar classification methods [Schuller et al. 2009].

**Table 3.1:** Distribution of different emotion classes in the EmoDB corpus. Even though the original experiment was designed for an equalised distribution, the set as cleared by human experts consisted only of the ones which provided a clear expression.

| Emotions | Samples | Rel. Samples in % | Time | Rel. Time in % |
|---|---|---|---|---|
| Anger | 127 | 25,8 | 5:34:77 | 24,1 |
| Boredom | 79 | 16,0 | 5:39.97 | 24,3 |
| Disgust | 38 | 7,7 | 2:08:73 | 9,2 |
| Fear | 55 | 11,2 | 2:03:70 | 8,8 |
| Joy | 64 | 13,8 | 2:43:89 | 12,1 |
| Neutral | 78 | 15,8 | 3:04:32 | 13,2 |
| Sadness | 52 | 10.5 | 3:27:43 | 15,4 |
| Total | 493 | 100 | 22:42:81 | 100% |

The corpus consists of 493 spoken utterance recordings, which are all annotated and based on a script repeated by each speaker. All of these recorded utterances were taken from a set of ten different sentences, where all were designed to have no textual indication of the contained emotion. The speakers themselves are equally distributed with five female and five male speakers, all in the age range from 21 to 35 years. The speakers are all trained actors and as such comparably similar in their strong and clear expression of emotions. Contrary to other datasets the emotional classes were chosen before the recording took place instead of being decided by the annotators. The recorded utterances were rated by annotators for emotional expressiveness and examples which were not recognisable by the annotators were removed from the dataset before publication. Specifically the recordings were rated on naturalness and recognisability, with a level of 60% and 80% being chosen as limit before being discarded respectively. This reduced an original set of 800 examples to the 493 mentioned. The used emotions were taken from the seven basic emotion classes of [Davidson 1994]. An exemplary listing of the recordings can be seen in Table 3.1. The high quality of the data is also due to its lab recording setup, instead of an in-the-wild method, and the technical high quality of the speaker expression for the different emotional classes, which is repetitive between different samples. These advantages allow for a clear observable correlation between extracted features and the annotated classes. Contrary to this, naturalistic speakers and open-microphone situations may lead to distracting outside influences.

For an alternative approach where this set can also be utilised as a two problem classifier, the emotions are mapped in this case to valence and arousal, allowing for an easy transfer of the classification setup. This is done in lieu of an expert annotation for these classes. While the recording setup was equally distributed, both the method of talking and the discarding of bad examples led to a skewed representation of training samples. This appears both on the discrete number of examples, but also in the cumulative time of for each example. In Table 3.1 it can be seen that disgust and sadness are considerably less present than anger. This can in turn lead to a bias in the achievable classification results.

### 3.1.2 eNTERFACE

Another example for acted emotions is the eNTERFACE corpus used in my research [Martin et al. 2006]. In comparison with the EmoDB set, the group of recorded speakers were all students instead of professional actors. At the same time the acting still enforces a similar style of expression, as well as an easier method of annotating the examples instead of interpreting naturalistic data.

Several other differences also distinguish the eNTERFACE from the EmoDB corpus, leading to different applications in my research. The dataset itself is multimodal in nature, specifically consisting of audio and video recordings of the participants, allowing the potential implementation of facial expressions as an indicator of the contained emotion. The sample size is greater compared to EmoDB, with 34 male and 8 female participants, 42 in total with a clear skewing towards the male speaker group. Compared to EmoDB the speakers are also more diverse in their expression, comprising of international students with different accents and mother-tongues. In EmoDB all speakers were native in the German language. As a method to build a baseline for the recording, all speakers were ordered to react in English, leading often to expressions in a non-native language for the participants. Similar to the EmoDB corpus the script again employed a variation of the seven basic emotion classes model with a slight adaption, specifically that neutral expression was removed from the set.

The biggest contrast is the induction of the emotion through a short story instead of a simple acting command through a script. Induction should ideally lead to a more naturalistic reaction, while still retaining the conscious aspect of the participant as they are aware of the recording and the knowledge that emotion is induced in them. The human experts in this case again measured the expressiveness and whether the given reaction was clearly part of the induced emotions or if the given recording was lacking a clear affiliation to a class. The recording itself was again taken in a lab environment, providing a standardised

**Table 3.2:** Distribution of different emotion classes in the eNTERFACE corpus, after the annotators removed all examples which lacked the necessary clear expression.

| Emotions | Samples | Rel. Samples in % | Time | Rel. Time in % |
|---|---|---|---|---|
| Anger | 200 | 17,1 | 10:51:56 | 23,7 |
| Happiness | 205 | 17,5 | 8:36:32 | 18,3 |
| Disgust | 189 | 16,2 | 8:44:16 | 18,8 |
| Fear | 189 | 16,2 | 8:47:28 | 18,8 |
| Surprise | 192 | 16,4 | 8:22:00 | 18,3 |
| Sadness | 195 | 16,7 | 9:56:20 | 21,5 |
| Total | 1170 | 100 | 46:30:24 | 100 |

setup for the speaker and the sensor equipment. The full set consists in this case of 1.170 recordings, with only the spoken part employed in my research. As can be seen in Table 3.2 there is still a certain skewing and bias in the number and duration of recordings between the different emotions, but this is comparably low when compared to EmoDB. Generally it presents a slight increase in complexity and naturalness of classification to EmoDB and allows for better measurement of the capabilities of a classifier than the former dataset for natural situations.

### 3.1.3 Speech Under Simulated and Actual Stress

The SUSAS dataset presents an intermediate transfer from the purely acted towards the naturalistic expression examples [Hansen & Bou-Ghazale 1997]. In contrast to the other datasets in this section, the SUSAS corpus is not one singular block of data but a collection of several different experiments and recordings. As it consists of both acted and non-acted parts it allows for training specifically these aspects which are similar between these parts. The main objective of the corpus is, as the name implies, the expression of stress through acoustic examples, specifically speech. The type of stress is in this case either workload stress, similar to the load experienced in the later Integrated Health and Fitness Corpus (iGF)-dataset in Section 3.2.1, but also so called situational stress such as fear or anger. The full set contains 32 speakers from which 13 are female and 19 male. The age of the participants is in the range from 22 years to 76 years which is comparably wide for datasets, providing in turn a wide variety of speaker styles. In sum there are 16.000 utterances in the whole set, which are itself divided into five individual sub datasets, an overview can be seen in Table 3.3. The general in-

**Table 3.3:** The different subsets were generated while the participants performed different task, specifically changing their speaking styles, performing single and dual tracking tasks, performing motion fear tasks and finally a simulated psychiatric examination of the participants. In the experiments performed, the chosen subset consisted from 3.593 segments all taken form the most natural part of the experiment where stress was induced though the movement of the rollercoaster.

| Recordings | Number |
|---|---|
| High Stress | 1202 |
| Medium Stress | 1276 |
| Neutral | 701 |
| Screaming | 414 |

duction was performed through tasks inspired by aircraft control situation under difficult situations, which were simulated on a rollercoaster.

## 3.2 Naturalistic Datasets

An alternative to the acted datasets presented in the last section is the use of naturalistic recordings. As the name implies the focus for this kind of data is that the expression of emotions, affects and user states is naturally generated, as similar as possible to a real world situation. As such, open-microphones and user recordings also fall into this category. While the focus on the acted data is its robustness of similar generated expressions and the clarity of the recordings, naturalistic data is comparably more diverse in its composition. As expressions change due to personal characteristics, as well as the strength and source for the expression of affect or emotion, the recorded examples also vary in volume and length. Additionally, the recordings are often no longer in a lab environment, which increases the influence of noise and irregular speaker behaviour, making the classification more difficult as a result [Dhall et al. 2016].

The examples for this type of dataset are the Integrated Health and Fitness Corpus (iGF), the Talk Run Speech Database (TalkR) and Munich Biovoice Corpus (MBC) and finally the SmartKom Database (SmartKom) and Restaurant Booking Corpus (RBC). Additional to the emotional examinations, these sets were also used for more indirectly measurable user states such as physical load or exertion.

**Table 3.4:** The iGF dataset information sheet as taken from [Tornow et al. 2016].

| | |
|---|---|
| Subjects | 65 |
| Gender | 20 Male / 45 Femal |
| Total Recorded Data | 105h 48min |
| Mean Duration | 97 min |
| Age | Min 50; Max 80; Mean 66 |
| Language | German |
| Annotation | Events, System Speech |

### 3.2.1 Integrated Health and Fitness

The iGF dataset is different in many ways to the previously presented sets [Tornow et al. 2016]. Contrary to a pure acted emotional set, the priority here was the naturalistic depiction of human mental states. The corpus was designed as a multimodal recording, containing audio recordings of utterances but also video recordings and bio-signals obtained by wearable equipment. For the research done by me, only the audio part was of importance. While the induction and expression of the user states was naturalistic, the recording of the audio samples was strictly controlled to remain similar between different participants and recordings.

There were 65 participants, with a composition of 45 female and 20 male participants, presenting a skewing towards the female speaker group. Contrary to the former datasets, the general age of the participants was also considerably higher with all speakers being between 50 and 80 years old, with an average of 66 years. A summary of the dataset can be seen in Table 3.4. Another important difference of the dataset was the implementation of both a Wizard-of-Oz [Kelley 1984] experiment and a cover story. The Wizard-of-Oz aspect was implemented by simulating a technical user interface for the participants to interact with, which in truth was externally controlled by a human researcher. The cover story in turn was a physical examination which induced different kinds of mental loads on the participants, which were not expecting a biased system with an agenda, but only an objective automated examination system. This complex implementation of the experiment allowed for a much higher naturalistic quality of the recorded reactions compared to acted or consciously induced states. This in turn allows for much more natural reactions from the participants. The induced user states were generated by requiring the user to create a physical test course which they had to fulfil, followed by an assessment by the system. The different states of mental load were then induced by (faulty) assessments and continuously more complex requirements in the planning process, which led to both increased stress and concentration from the participants. After each stage of the test the participants were relaxed both by a cool-down period and a relaxing musical induction, allowing the participants to start from an approximately neutral condition. Additionally, the level of mental load was continuously heightened, such that even an overlapping of the inductions would still follow the script of the experiments.

### 3.2.2 Talk Run Speech Database and Munich Biovoice Corpus

Two different sets were employed for the work with physical load detection through human speech, the TalkR [Truong et al. 2015] and the MBC [Schuller et al. 2014a]. Both sets consisted of examples which contain participants performing real physical tasks to induce physical load. As physical stress is otherwise hard to simulate, this is necessary for a sensible classification task. The TalkR set was taken from 21 participants, which are separated into 15 female and 6 male participants respectively. The age of them ranged from 20 to 31 years of age. The primary aspect of this corpus was that the recorded utterances were full sets of sentences, which were recited during a physically demanding task. As to establish a ground truth, all speakers had to display a heart rate of 172 to 198 BPM during the recital. As the set contained both recording in English and Dutch as examples, the probability of a context sensitive interpretation of the system is comparably low. The full set was comprised of 250 separate recordings with a length of roughly 85 minutes for the full set.

The comparison and extension set for the experiment, the MBC consists of only 19 participants with a separation in 4 female and 15 male speakers. Contrary to the first examples, the speakers were only instructed to intonate singular vocals during the experiment and one exemplary sentence taken from the former TalkR set. The length of this set is nonetheless higher, with the full set consisting of 74 utterances, and 630 recorded vowel expressions. This increase appeared because the recordings were done before and after a set of physical exercises, with a designated minimal heartrate of 90 BPM.

**Table 3.5:** The frequency and distribution of User States in the SmartKom corpus taken from [Steininger et al. 2002]. The modified column depict the data as used in experiments presented in Section 6.3, where the data was recut into more equalised durations.

| User-State | N | % Number | % Duration | Modified |
|---|---|---|---|---|
| Neutral | 1253 | 43.7 | 71.6 | 2179 |
| Pondering | 689 | 24.0 | 14.0 | 643 |
| Joy | 370 | 12.9 | 7.2 | 284 |
| Anger | 205 | 7.1 | 2.8 | 220 |
| Helplessness | 182 | 6.3 | 3.3 | 161 |
| Surprise | 99 | 3.4 | 0.6 | 70 |
| Unidentifiable | 72 | 2.5 | 0.6 | 266 |
| Total | 2870 | 100 | 100 | - |

### 3.2.3 Smartkom

The SmartKom multi-modal corpus [Steininger et al. 2002; Wahlster 2006] contains naturalistic affects within a Human-Machine Interaction (HMI) environment. The system responses were generated by a Wizard-of-Oz setup, recorded under studio conditions. The database contains multiple audio channels and two video channels (face and body in profile posture). The primary aim of this corpus was the empirical study of HMI in a number of different tasks. It is structured into several sessions. Each session contains one conversation and is approximately 4.5 minutes long.

The background story for this dataset was the use of a technical interaction prototype, which was in truth controlled by human operators. The participants were given a range of different tasks, which they had to accomplish using this system. As such it allows for a (perceived) natural HMIto take place. As the external instruction were minimal, the participants were allowed to develop their emotional reactions naturally during the interaction, while the lab-like environment lead to a high technical quality of the recordings. The distribution can be seen in Table 3.5. The data was labelled by three instances of independent human annotators to ensure a correct allocation of the user states, further ensuring a high quality of the generated data.

**Table 3.6:** The RBC dataset information sheet as taken from [Siegert et al. 2019].

| | |
|---|---|
| Subjects | 30 |
| Language | German |
| Gender | 10 Male, 20 Female |
| Age | Min 18; Max 31; Mean 23.97 |
| Recorded Data | 5h 37 min |
| Mean Duration | 193.6 sec |
| Utterances | 4835 |
| Annotation | Utterances, Transcriptions, Context |

### 3.2.4 Restaurant Booking Corpus

The RBC is a dataset consisting of acoustic recordings between human speakers and two different technical assistant systems and a human interaction partner, where the human speakers attempt to book a place at a restaurant under certain constrictions, with the aim to distinguish the directedness of the speakers either towards a human or a machine, based on the voice. The dataset was recorded from 30 German speaking students, and consists of 10 male and 20 female speakers. They provided 90 dialogues with different levels of constraints, such as different times, accessibility or food choice. All dialogues were annotated by humans to ensure the textual similarity between the device-directed and the human-directed speech. An overview can be seen in Table 3.6. As the experiment was done in a Wizard-of-Oz setup with a cover story, the naturalness of the generated recording is high. The recorded data was then evaluated by human annotators, which showed an Unweighted Average Recall (UAR) of around 60% for German speaking and around 54% for non-German speaking annotators to correctly identify the directedness of the speakers.

## 3.3 Summary of the Chapter

In this chapter I presented the used datasets in my thesis, with the main attention on the baseline and benchmark examples for the respective categories. Both acted and non-acted or naturalistic data provide necessary aspects for the training of machine learned classifiers presented in the next chapters. As the achievable results, such as accuracy and robustness, are closely dependent on the chosen dataset, it is equally important to ensure sufficient and qualitative data for the system. Together with the last chapter this concludes the presentation of the employed tools in this thesis. Beginning with the next chapter these will be used in my original research in the applicability of a more advanced level of assistance system.

Chapter 4

# Significant Feature Identification

## Contents



The first relevant part of a Human-Machine Interaction (HMI) capable system, such as the conceptualised assistant system presented in this thesis, is the implementation of an interface between the external and internal space of the machine [Gong 2009]. As seen in the concept illustration in Section 1.2, this equates functionally with the first two layers of the assistant system. In this chapter, the specific transfer between the real (outer) world and the technical (inner) world is treated in greater detail.

During the process of converting between these two frames of representation, especially in the case of an advanced and complex technical system employing machine learning solutions, traditionally a great amount of data needs to be gathered and processed [Adadi 2021]. This amount of data can counterintuitively lead to decreased performances in machine learning applications, when processed without further considerations [Bellman & Kalaba 1959]. Generally it also leads to higher computational loads, or slower processing times. To reduce these negative effects, pre-selection and pre-optimisation of the data is helpful and sometimes necessary, which made it into one of my first greater areas of research.

This chapter specifically concentrates on the possible and novel methods employed in my research to reduce this negative influence of raw data collecting by employing feature selection and optimisation methods. This was necessary, as in the speech based experiments performed by me, these effects were amplified by the highly individualised data sources, as speech styles vary greatly between different speakers [Siegert et al. 2013].

The topical structure for this chapter here is as follows, beginning with the typical applied methods and their problems, which are shortly explained in Section 4.1. This is followed by a novel approach to find and employ the most significant numerical features, as measured by their influence and impact on a later classification tasks, specifically in case when examples are hard to distinguish due to overlapping feature spaces, in Section 4.2. A further experimental approach is a novel form of visual feature selection applied on transformed acoustic data and its specific advantages, as examined in Section 4.3. In Section 4.4 the results of this chapter are summarised and presented in context of the overarching work.

The research cited here are mainly based on the publications of my colleagues and myself in [Weißkirchen et al. 2017; Weißkirchen et al. 2018; Weißkirchen & Böck 2018; Egorow et al. 2019] and [Weißkirchen et al. 2020b].

## 4.1 General Feature Pipeline

The first step for a functional HMI system, specifically one based on machine learning, is an effective feature selection and feature reduction. This is primarily done to achieve better results for the following classification tasks [Chen et al. 2020], but also to achieve sufficient results in a sensible timespan [Sevilla et al. 2022]. Parallel to the work examined in this thesis, this interface architecture consists of the whole pipeline which transforms real world information into machine interpretable data. While in this thesis the main attention is on sound based information, such as user generated speech and the research on the pipeline is generally constrained on data representation, this specific section is designed to give a small outlook beyond this scope. When examining the full process, one has to look at the environment, in which the system operates, or specifically the real world data which represents the area of application for the system. As shown in Chapter 3, the datasets which are used for training of technical systems can often be distinguished between real world recordings and acted situations. This not only influences the way in which the data is constituted itself, but also has a technical influence on the quality and amount of information [Jürgens et al. 2011]. While training is often done on acted data, the requirement is often that a transfer of abilities into the real-world is possible. Environments where both humans and HMI-capable systems coexist and work together in this real world are in turn often called a “Smart” environment, such as smart homes or smart factories [Asadullah & Raza 2016], declaring the general complexity of establishing such an application.

In such an environment there are not only a variety of technically controlled appliances and the overarching control system, as presented by the assistance system, but very importantly also a wide array of sensors. These sensor arrays provide the first technical step of the environmental awareness and, in turn, the generation of usable features. The scope of possible research into sensors and their application is consequentially wide and variant [Braun et al. 2014], but as in the proposed architecture the main priority lies in voice based interactions and sound based information extractions, the typical exemplary implementation is that of microphones. Further sensors which are usual in home-like environments are of the visual and/or general movement detecting kind, while more personally integrated systems often also comprise of bio-signals, which are detectable by systems like “smart watches” or further wearable implements [Jat & Grønli 2022]. This not only allows for an easier integration in the typical HMI process, most often used in the current state-of-the-art as seen in Section 2.3.1, but also reduces the level of conceived intrusion into the lives of the users [Hayashi & Ruggiero 2020], which are in turn an important aspect for the subjective quality of life.

Beyond the necessary step of changing the available analogue information into digital representations, such information has also to be pre-processed further to improve its expressivity. This takes the form of feature extraction (such as the different audio feature sets, seen in Section 2.1) or the normalisation of data into certain value ranges to stabilise the output of different sensors against outlier results [Knight et al. 2019].

A system integrated into the typical lifestyle of a user would require a great number of such sensors, as these directly influence the awareness level the system can generate from its user and their direct environment. This also leads directly into the main topic of this chapter, the feature selection processes themselves. Without, intelligently, reducing the amount of interpretable raw data into a smaller size, while at the same time not losing relevant information, technical systems are either overwhelmed on a computational level or even lose capabilities by introducing too much information in the training process [Ying 2019]. Such effects are explained in greater detail in Section 2.2, especially their effect on classification tasks, which are often a significant reduction of the achievable results. This effect is also explained under the term "Curse of Dimensionality", and as such is important during the implementation of a complex system [Bellman & Kalaba 1959].

The general choice of which features to use, specifically for voice based tasks, is traditionally to be using features which are designated by experts to contain the relevant information, or which are proven to work in similar natural interactions. In case of voice based systems, this primarily influences the choice of the feature sets, as presented in Section 2.1, such as Opensmile's emobase (emobase) or Opensmile's emolarge (emolarge) [Eyben et al. 2010]. While these are still of a great size, they are a much more efficient representation of the information contained in the voice than the simple digital audio recording format by itself, which are not that accessible for machine learning systems to process [Natsiou & O'Leary 2021]. Worth mentioning here, are the so called Mel-Frequency Cepstral Coefficient (MFCC) features which are digital representations of the way human listener receive sound signals, instead of the way a microphone would record a signal [Zheng et al. 2001; Dumouchel et al. 2009], and which provide an often used standard input for classification tasks because of its closeness to the human perception.

The usage of feature selection is done traditionally in a variety of ways, which includes mathematical methods, which are also examined on a basic level in Section 2.2.1. For completeness, they will be mentioned only shortly here as well, in as much as they were used in my research where the regular approach was sufficient. Also shortly mentioned is the closely connected idea of dimension reduction, which also reduces the amount of processed features, by combining them into a more information rich representation.

These methods are usually variants of the Analysis of Variance (ANOVA) [Girden 1992], which measures the probability that a certain variance of a variable is part of a certain group representation. This represents the basic idea behind a feature selection, where only the variables are used which are relevant to distinguish between several classes. In the case of my research these were often used in classifications tasks, with the groups representing the different classes which need to be separated automatically.

A classic example for the employed dimension reduction is the Principal Component Analysis (PCA) [Pearson 1901]. This method aims to reduce the dimensionality of the data, and in turn the amount of input features necessary, by mapping the data points onto the primary components, which maximises the variance inside the data set as a whole. With that the machine learning methods can easier separate the feature space for the different classes. This method was used for example in [Egorow et al. 2019].

## 4.2 Overlapping Feature Identification

A typical approach in classifying speech information with machine learning methods is the use of numerical values, representing the different information features collected [Schuller et al. 2004; Böck et al. 2010; Sezgin et al. 2012]. This is a result of the kind of features used, which can often be easily represented and quantified such as the recorded tone frequency or the loudness in decibel [Ververidis & Kotropoulos 2006]. As explained generally in Sections 2.1 and 4.1 before, this data collection may easily encompass such an amount of data that the fully trained solution from the learning algorithm is worse than it would have been with less available data or with some possible features missing, which is a known problem in this field [Vinciarelli et al. 2009]. The main question for this section follows as such, how can a technical system decide which data is the most relevant for a given recognition task? Additionally, are there alternatives when the available data is more complex in its composition than in comparable situations?

The general applied solution in this case, which is possible when using numerical inputs, is the use of statistical pre-selection methods, such as the mentioned ANOVA [Girden 1992], to assign which input features have a greater influence on the final classification result. A specific problem, such as the use of paralinguistic features to recognise the current emotional or mental state of the user would reduce the potentially necessary amount of data considerable, using these methods. As mentioned in Section 2.1 the amount of features normally extracted from sound, specifically voice, can easily be in excess of 6.373 or even 6.552 functionally different features when using Interspeech ComParE 2013 (ComParE) or emolarge feature sets respectively, as an example [Schuller et al. 2016; Eyben et al. 2010]. These already present a specialised and reduced feature set for sound and voice data. It is also known that the used amount of data can be significantly reduced for specific tasks [Böck et al. 2010; Xu et al. 2014] even while retaining similar results, as the relevant information varies based on the application.

Other experiments [Xu et al. 2014] have also shown, that the optimal feature set is often quite different depending on the appointed task, which in turn makes the generation of one optimal feature set for all voice related tasks improbable, even though there are some features which provide important information for a variety of tasks, such as the mentioned MFCCs [Zheng et al. 2001; Tahon & Devillers 2016].

An additional aspect of this problem is a result of the high individuality of expression in different speakers. This individuality, which results from the biological and social background of the individual speaker, influences the way of talking and vocal expressiveness immensely [Brown et al. 1991; Resseguier et al. 2016]. This leads to a high complexity for a technical system, as typical statistical examinations over all available speakers would show no singular group of features which adequately maps all or even most of the speakers to a specific result. As the regular ANOVA requires a weighted distribution, a sufficient application of such a method becomes less probable. This results in an effect we called "overlap" in our research [Weißkirchen et al. 2018], which describes the effect, that one speaker with a certain user state sounds, more precisely measures, similar to another speaker with a different user state. The resulting question now is:

> Is there a possibility to designate relevant features even when the overlap inhibits the usual methods, and finally if the found features can be useful for later machine learning tasks?

An example for this is an experiment, which will be explained in greater detail in Chapter 6, and concerns itself with the classification of the mental load state of a user. These states, which are not easily distinguishable in the feature space, are in turn also hard to distinguish by machine learning methods. As such, a simple statistical feature selection can lead to either mediocre results (close to the random guessing) or necessitating greater amounts of training data, more complex architectures and more training time for good results. The idea behind this experiment was then to recognise alternative forms of recurring patterns between these different states, which can then be used to design specific extractable features instead of relying on typical speech features which hinders the classification process due to their overlapping nature.

An examination was done by us [Weißkirchen et al. 2018], concerning this specific problem of mental load in the sense of high stress during HMI, which led to the realisation that even with this high individuality of the speakers, the feature changes for the individual speaker between the mental load states designated by us (high and low) were similar. As such, even a few features were sufficient for deciding the mental state of a singular speaker, while at the same time proved inadequate for a general application.

Recognising this difference between a general applicable feature and a feature which is important for a specific speaker, or generally a specific situation, is one of the basics behind the idea of a (personal) adapted system, as shown in Section 5.3. To solve this problem, we designed a different approach to designate important features as an extension to the regular statistical methods.

Our research was done on the Integrated Health and Fitness Corpus (iGF)-Corpus [Tornow et al. 2016], which induced different states of mental load on the participants by a mixture of timed planning tasks with either positive of negative assessment by an (assumed) observing technical system. An important distinction has to be made at this place, while the current terminology often describes “mental load” in the context of gender specific stress during housework, in our research it described the gender independent level of mental stress when solving complex tasks under time constraints, the other meaning was not apparent to us at the time of the original research.

The available (audio-) recordings provided examples for the participants being in different levels of mental load and engagement, from which we used the two stages with the most “extreme” induction of either “Underload” (very slow decision making required with general positive assessment) and “Overload” (fast decision making required with general negative and confrontational assessment), as explained in Section 3.2.1 with further information on this dataset in particular. The dataset and task were chosen especially because of its comparably high complexity of connecting a highly subjective expression of the inner user state with the recorded speech during a HMI.

When examining all individual occurrences, in this case of the 65 different speakers, instead of finding the statistical most important features based on the whole group, the aim is to find the features which most significantly change between the classifiable results for each singular speaker. The full set of available features were then taken from the emobase feature set, as presented in Section 3.1.1, which contains, among others, MFCC and Linear Spectral Pair (LSP) as well as their derivatives values with 988 features in total.

Additionally, instead of taking the values as such, we observed only the changes between both states of the mental load, eliminating the underlying individuality of the speaker dependent baseline for each feature, e.g. a speaker with a natural higher or lower voice. The resulting change value of the features is a very potent feature by itself, but as the measurement requires pre-knowledge of the baseline value before the change, which is fully based on the individual speaker, it is nothing a technical system could extrapolate easily on its own, without also knowing the speaker beforehand.

To assure that no random or biased conclusions are drawn, we employ the Kruskal-Wallis test [Kruskal & Wallis 1952] over the measured differences to assert if they are part of inherent behaviour due to mental load or random occurrences. This method by itself is used to determine if several samples belong to different distributions, in this specific case if the change of a feature belongs to different occurrences of a speaker. The idea behind it is to prove the opposite, specifically that a change of the feature is inherent to the change of the mental state.

The features can then be sorted by "significance"-level, specifically the scientific meaning of the term. For this the resulting p-value of the test has breakpoints below the probability of $p<0.05$ of being a significant feature if it would only change randomly for all the speakers, with a $p<0.01$ to be highly significant occurrence and $p<0.001$ to be a most significant random occurrence, which in turn very strongly implies a non-random change of the feature. Resulting from this examination, a ranking of the features can be done, such as seen in the Table 4.1, which shows the counted occurrence of the different levels of significance over all the examined speakers.

As the results are added over all the speakers, changes which appear for all or most speakers prove the importance of the feature itself. But while these features are highly relevant in the distinction of the user state of all the speakers individually, they are, as mentioned, not sufficient for a general classifier over all subjects at the same time (or by the same classifier). To use this relevant aspect of information for a solution, my colleagues and I propose a layered classifier in [Weißkirchen & Böck 2018]. There, the problematic task of using the overlapping features is separated into smaller and easier classification tasks, roughly forming an equivalent to a deep learning architecture by layering functional steps in its design. The system would thereby solve two or more separate decisions, with a comparatively higher level of accuracy and recall, while needing much less data and computational time for each, compared to when the whole task would be done in a singular classifier.

This is done, by finding smaller groups of speakers which share their characteristics more closely with each other than with the rest of the group. In the specific case this was examined for biological sex and rough age groupings, as

**Table 4.1:** A list of the features with the most significant difference for the measurement of mental load, taken from [Weißkirchen et al. 2018]. The aggregate was done over all 65 speakers on how often the difference between low and high mental load were in the field of most, highly and regular significant differences. All 933 features of emobase were available, and only the presented ones were necessary for a adequate classification. For further explanations of the feature names, see Section 3.1.1.

| Significance | Most Sig. | Highly Sig. | Sig. |
|---|---|---|---|
| pcm_intensity_sma_de_amean | 50 | 57 | 59 |
| pcm_loudness_sma_de_amean | 48 | 52 | 57 |
| pcm_intensity_sma_de_linhregc1 | 47 | 51 | 56 |
| lspFreq_sma2_range | 47 | 53 | 56 |
| lspFreq_sma1_range | 47 | 54 | 55 |
| mfcc_sma_de1_range | 45 | 53 | 59 |
| lspFreq_sma_de0_maxPos | 44 | 52 | 56 |
| pcm_zcr_sma_de_maxPos | 43 | 52 | 58 |
| pcm_zcr_sma_de_range | 43 | 49 | 58 |
| pcm_zcr_sma_range | 42 | 51 | 59 |
| lspFreq_sma_de5_amean | 41 | 52 | 57 |
| pcm_loudness_sma_de_amean | 41 | 53 | 57 |
| lspFreq_sma7_maxPos | 40 | 49 | 55 |
| lspFreq_sma_de7_range | 39 | 47 | 50 |
| voiceProb_sma_maxPos | 39 | 45 | 51 |
| mfcc_sma2_range | 39 | 48 | 54 |

both characteristics strongly influence the general way voice is created by the body [Sidorov et al. 2016; Pisanski et al. 2016; Taylor et al. 2020]. As can be seen in Table 4.2, this changes strongly within and across groups. In comparison, the correlation for the full feature set between all speakers is 30%, which improves by ca. 1% when only compared within a group of speakers with the same sex. The answers which can be taken from the research and its examination are as follows: There exist features of high relevance, even beyond the scope of which can be found by a typical ANOVA, but these are hard to integrate into a machine learning approach. To alleviate this, a problem can ideally be separated into smaller but overall easier problems, often allowing the inclusion of former and expert knowledge. Additionally, as the singular tasks gets smaller and easier observable, the decision making process and location of potential errors become easier to trace. This is especially important, when compared to the otherwise

**Table 4.2:** Presentation on how features correlate for different age groups in the designation of mental load, taken from [Weißkirchen et al. 2018]. The age groups only present rough grouping of ca. 10 years range, as the dataset included only elderly subjects. Higher correlation presents that changes in the features for the speakers was similar. Low correlation shows changes in different features and different values occured.

| | | **Men** | | | | **Both** | **Women** | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | "Young" | "Middle" | "Old" | All Ages | | All Ages | "Young" | "Middle" | "Old" |
| **Men** | "Young | **1.00** | 0.82 | 0.93 | 0.92 | 0.93 | 0.93 | 0.74 | 0.98 | 1.0 |
| | "Middle | 0.82 | **0.72** | 0.87 | 0.79 | 0.78 | 0.77 | 0.67 | 0.81 | 0.82 |
| | "Old" | 0.93 | 0.79 | **0.87** | 0.87 | 0.86 | 0.86 | 0.69 | 0.91 | 0.93 |
| | All Ages | 0.92 | 0.79 | 0.87 | **0.89** | 0.87 | 0.87 | 0.69 | 0.91 | 0.92 |
| **Both** | | 0.93 | 0.78 | 0.86 | 0.87 | **0.87** | 0.87 | 0.70 | 0.92 | 0.93 |
| **Women** | All Ages | 0.93 | 0.77 | 0.86 | 0.87 | 0.87 | **0.87** | 0.70 | 0.92 | 0.93 |
| | "Young" | 0.93 | 0.67 | 0.69 | 0.70 | 0.70 | 0.70 | **0.59** | 0.73 | 0.74 |
| | "Middle" | 0.98 | 0.81 | 0.97 | 0.91 | 0.92 | 0.92 | 0.73 | **0.97** | 0.99 |
| | "Old" | 1.0 | 0.82 | 0.93 | 0.92 | 0.93 | 0.93 | 0.74 | 0.99 | **1.00** |

unknown "blackbox" process which is often a concern for machine learning applications, as explained in Section 2.2.1.

## 4.3 Visualistic Features

When using the more complex classification methods, such as Convolutional Neural Network (CNN) architectures or similar specialised solutions, some of the typical feature selection methods are no longer applicable or at least less efficient. Specifically, some of these advanced architectures are optimised to use visual input formats, typically in the form of colour images, practically three-dimensional value arrays or tensors. The technical most relevant difference in that case is, that information is not only contained in the value of a feature but also its location in relation to other features [Islam et al. 2020]. The strength of CNNs for visual input information is explained in greater technical detail in Section 2.2.1. Following this trend of advancing research, CNN systems also present one of the most capable currently available implementations, concerning the ability to classify even complex task, as well as providing unique methods of supervising and understanding the training process. A complex assistant system employing machine learning methods in an open, real world environment could benefit from these strengths and the further developments emerging from them. As such the main questions in this section will be what general approaches are possible to employ these methods in the area of mainly speech based applications, if there are any objective indications, based on the available features, which level of complexity the system should employ, and if there are further advantages in the usage of visual representation of data beside the use in CNN or similar architectures.

### 4.3.1 General Visual Features

The usage of visual representations for acoustic signals, or in general of non-visual signals, allows for the employment of different methods to identify the significant input features besides the mentioned ANOVA ones. Instead of simply measuring the occurrence of values of each input feature, which is done for the typical statistical methods, the location and relation of the values inside the input array to each other are also important, as the convolutional process of a CNN processes these inputs through its “kernel” architecture [Krizhevsky et al. 2017]. This allows for objects and patterns in an image, for example eyes, nose or mouth to contribute to an overarching face detection task, as an example [Paul et al. 2014]. When changing the data representation into such a form, employing these new possibilities could potentially be beneficial also to other classification tasks.

The simplest method with images to measure the influence of certain parts on a classification task, is by occluding information from the input and examine the changes in the classification result [Zeiler & Fergus 2014]. This is especially conclusive for real images as the areas can ideally be directly identified, for example the influence of the eyes on a face detector [Izhar et al. 2023]. Such a method can remove the possibilities of learning the false object or area, when this would otherwise inhibit the generalisation process during machine learning. See Section 2.2.1 for an example of a situation where a system learned the background instead of the relevant object, leading to good classification results, while at the same time not providing a good classifier. This is particularly useful to deduce biases in the training data, as it can happen when the training data was all taken from the same source, such as in [Yu & Eng 2020], while the test data is coming from another source. As a negative point of this method, it is only helpful in situations where the information is grouped in close proximity to each other, such that the occlusion may obstruct the faulty pattern fully in one step. Unfortunately, by employing this method to more unusual data types, such as spectrograms of voice or other more dimensional representations of originally arrays, this is not as conclusive.

Another, more complex, method, employs the layered architecture of CNN to its full effect. As the convolutional process is not only applied on the first (image) level, further information may be grouped over bigger distances in the applied input. As an example from an (idealized) real application, as seen in [Krizhevsky et al. 2017], this can lead to edge detection in the first step, eye detection in the second and face detection in the third. An occlusion experiment would only work sufficiently in the first, at most in the second stage.

The idea instead was to use the training method in a reverse application; as such the system tries to recreate an original image, matching or generating a perfect representation of the chosen class at the end. The method resembles the deep dreaming approach [Szegedy et al. 2014a] also presented in a more technical detail in Section 2.2.1, but with a stronger emphasis on the specific classes. This method has the advantage of producing potentially full images containing all the information already trained for a specific class instead of only one partial aspect of the image. An example of this can be seen in Figure 4.1 as applied to a spectrogram which was used to train an emotional classifier. The appearing image develops into an amalgamation of the used spectrograms during the training of the specific class, where frequently occurring patterns appear stronger on the final image. A caveat of this is, that the resulting image also includes all other aspects of the spectrogram which were employed during the training and may represent unwanted biases occurring from the chosen training sets.

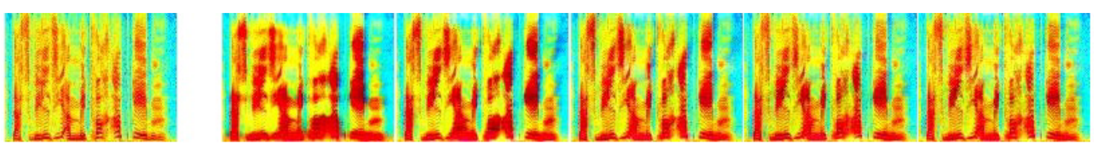


**Figure 4.1:** Examples of the Deep Dreaming like method to induce the relevant features into an image. The image on the left is the "seed" picture necessary to start the process, potentially the system should work on any noisy image, but this would need a greater amounts of training data, as well as a longer time to develop. The images on the right were taken after each 100 iterations, with the emotion "anger" as target. One can see that the areas first overlap, and then enhance certain visual structures on the original image. Images taken from [Weißkirchen et al. 2017].

### 4.3.2 Comparing Visual to Numerical Features

Given that a switch from the acoustic representation to a visual is possible and, as mentioned, also allows for some computational benefits, like the use of CNN or data augmentation, the decision has to be made by the system designer as to when this change is useful. While a purely visual input allows for the generation of "heat maps" of relevance and are easier to comprehend trough visualisation, purely numerical feature methods employ the generally more proven and established ANOVA based approaches, as presented in Section 4.3.3. The occurring question here is, can the availability and type of features themselves influence whether the transformation into a visual representation is useful?

An experiment where both approaches were employed was done by my colleagues and myself in [Egorow et al. 2019]. There the same datasets were classified using either a numerical input with feature reduction, as well as a comparable experiment employing spectrograms and a CNN classifier. The experiment itself was done on the Talk Run Speech Database (TalkR) [Truong et al. 2015] and the Munich Biovoice Corpus (MBC) [Schuller et al. 2014a] sets of data, which contains recordings of subjects under either low or high physical load, with either 21 or 19 subjects respectively. The aim of the experiment was to classify which state the subject had, based on the recorded voice. Further information of the datasets can be found in Section 3.2.2, while the classifying results are discussed more in-depth in Chapter 6.

The data was prepared in two different ways, the first was based on the typical numerical value input, which we employed with a Feedforward Neural Network (FFNN) architecture of machine learning. For this, the extracted features were taken based on the Interspeech 2011 Speaker Challenge [Schuller et al. 2011], which proposed a set of 3.396 features based on their ability to recognise the user state of a speaker. These included energy and spectral related features, and were also used in the specific Physical Load challenge from 2014 [Schuller et al. 2014b] where physical load was the relevant user state.

The second method was using the capabilities of a CNN system as explained in Section 2.2.1. The chosen input format was to use a grayscale spectrogram, which instead of using three channels for colour employed the other available channels as the first and second derivate over time for the values of the grayscale image. As mentioned in Section 2.2.1, a CNN system is originally designed for visual input, but can theoretically work on other inputs as well, with the architecture itself being designed to work with input matrices'.

The first method processed the data with the help of a bottleneck network, similar in idea and implementation to a sparse auto-encoder [Schmidhuber 2015]. Such an approach reduces the number of available transmittable values, while at the same time retaining the general information level between each step. This ideally removes redundancy in the available data and generates a more effective representation of the original information.

As a short by-way explanation, autoencoders ensure the completeness of the transferred information by reconstructing the original input signal. In the case of our bottleneck approach these reconstruction steps were cut off after successful training of the autoencoder. The remaining net was then instead fed into the classifying FFNN architecture, as mentioned. This provides an option to produce feature reduction on an automated scale. The autoencoder itself can be trained before such a system would be implemented, after which the relevant classifying system could operate on much smaller data amount and feature sizes. This would effectively reduce the necessary calculation time, especially in complex tasks.

The other approach used the abilities of the CNN architecture for its feature selection and optimisation task. As mentioned in Section 2.2.1, the strength of CNNs is their ability to effectively optimise the data flow when using images as an input, as this is part of the general convolutional processing steps, as shown in [Krizhevsky et al. 2017]. These so called kernels work on an area (or volume in the case of 3d-arrays) of the input at the same time, retaining information coded in the relation of the image values to each other, even over multiple processing steps. Contrary to most other machine learning methods, CNNs are capable of producing their own filter functions this way [Krizhevsky et al. 2017]. This leads to much improved results for most image based tasks, to which this experiment belongs to. While convolutional filters would, based on their implementation, reduce the necessary data size, there is a distinction to be made. The resulting convolution of an image is smaller (dimensionally and data wise) compared to the original input. It is also traditionally done several times over each convolutional step, generating a "deeper" representation. Alternatively, it can also be understood as the result of several filters applied to the same area of an image. While the result itself is smaller than the input, the amount of filters is often higher than the "depth" of the original image. This approach as such is closer to the idea of intelligently interpreting the input features and not so much for the reduction of data during the task.

The aim of the experiment was now to compare these two approaches for a potentially clearly advantageous option. This proved highly dependable on the available data. While the experiment in practice proved better for the FFNN option, this was only the case when the system was trained on one dataset and then employed on the other one, proving the generalisation capabilities of this

**Table 4.3:** Comparison of a FFNN and a CNN classifier for physical load detection on the TalkR and the MBC datasets. The given values are the Unweighted Average Recall (UAR) and are taken from [Egorow et al. 2019].

| System | Females | Males | Overall |
|---|---|---|---|
| | TalkR | | |
| FFNN | 83.4 | 69.7 | 79.22 |
| CNN | 81.8 | 73.2 | 79.52 |
| | MBC | | |
| FFNN | 57.7 | 62.3 | 60.71 |
| CNN | 55.9 | 55.5 | 55.69 |

comparably simpler solution. The results of the classification compared between these two methods can be seen in Table 4.3, while the architectures themselves are part of further examination in Section 5.2.2 and Section 6.2.2 respectively. Conversely for the alternative method, the CNN architecture requires greater amounts of training data to provide the baseline system with enough material to generate the optimal filter capabilities. Secondly, while the training of a CNN lasts considerable longer than a FFNN of similar depth, the computational time necessary for the process afterwards for the classification itself is roughly of a similar length.

As a general decision point, for a problem with high amount of training data and without much prior knowledge of usable features and their interpretation, a system such as a CNN can provide the necessary complexity as well as the ability to train its own optimisation. A problem for which the amount of available data is relatively small, but for which the classifying complexity can be assumed to be comparatively low, the better generalisation of a FFNN with added feature selection and reduction steps is better.

### 4.3.3 Visual Keypoints

Besides using the methods typically associated with CNNs, the step from converting the feature arrays into image representations also allows for different realisation methods. The methods, mentioned before, assume a visual representation where the system is either employing the feature values with the added benefit of local feature groupings as in Section 4.3.2, or the typical CNN classifier, roughly representing a trained technical expert looking at an image, here a spectrogram.

An additional avenue is the option to view the task as a purely vision based problem and employing the options connected with computer vision algorithms. In [Weißkirchen et al. 2020b], we proposed exactly such an idea, by using a spectrogram based classifier with the speciality of using computer vision algorithms to detect the relevant points of interest for the different emotional states of a speaker, while employing a Support Vector Machine (SVM) for the classifying task itself.

This would combine certain aspects of the local groupings inside the visual representation, which would be missing in a pure feature array representation, while also employing the much less computationally complex methods of a SVM instead of a CNN approach.

The key of determining the relevant points with computer vision algorithms is the ability to discern repeating patterns between examples of the same class, in this case a specific emotion, while foregoing the complex learning algorithms of a CNN. This promises a better generalisation ability, while requiring less data, to achieve satisfactory results. The requirement for such a system to effectively work, is the ability to detect these patterns, especially when certain variations are a natural part of the data, specifically in the way individual idiosyncrasies of the speakers influence the resulting spectrogram.

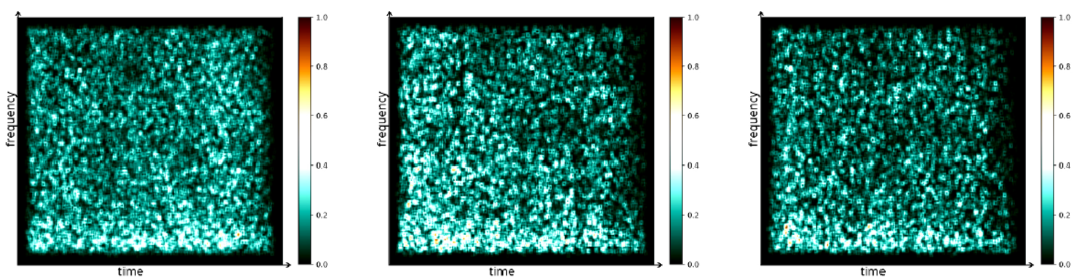
A visual detection algorithm achieves pattern recognition by comparing points and their surrounding between several other examples. In a spectrogram this reflects certain patterns of the frequency and time which re-appear for most examples of a class. As it is not bound on either axis (frequency or time), neither higher or lower frequency, nor earlier or later appearance of the pattern in the spectrogram are relevant. Converted into speech, this would present different spoken utterances or speakers with a different baseline frequency in their voice.

More complex methods are capable of rotating, and sometimes stretching, the compared pattern. In typical images this would represent objects which are closer or farther away from the camera, while in our visualisation of speech it instead represents small changes in the expression not covered by the former approach.
The used dataset was the SmartKom Database (SmartKom) multi-modal corpus [Steininger et al. 2002], from which only the audio recordings were used. It covers 3.828 samples and is received from an HMI, closely resembling the requirements for an assistant system, further information of this dataset can be found in Section 3.2.3. The distinguishing classes we examined were the typical seven emo-

tions, and a second experiment was done using the more compact valence-arousal classification, see also in Section 2.1.1.

As there is a variety of different visual extraction algorithms, we decided on a method being comparably fast and capable of working with image transformations. The chosen method was Oriented FAST and rotated BRIEF (ORB) [Rublee et al. 2011], an improved variant of the Binary Robust Independent Elementary Features (BRIEF) [Calonder et al. 2010], which is capable of providing these comparisons while including the ability to perform the necessary image transformations. Both methods were also compared with each other, which is examined further in Section 6.3 for the achieved classification results.

As spectrogram images are comparably noisy, the algorithm was applied after a Gaussian filter was used, to reduce the sensitivity of the system for it. The key point detection was done by employing the intensity threshold between the central pixel and its surrounding with a 9 pixel width. When comparing the




(a) Anger (b) Helplessness (c) Neutral

**Figure 4.2:** Heatmaps of key points density for different emotion classes, taken from [Weißkirchen et al. 2020b]. One can see different areas of interest. What is not directly visible, is that the type and pattern of keypoints also is an important part of the detection making process.

average location of the key-points over all representations of certain classes, a type of heat-map develops, as seen in Figure 4.2, representing which areas of the spectrogram are important for each class. The existence of the key-points in itself, and their locations, can then later be used for classification tasks, as presented in Section 6.3. The necessary amount of key-points could be as low as 10 compared with the original full size spectrogram, presenting another option to reduce and pre-select the available data. In comparison, similar experiments which used the much larger emolarge feature set with 6.558 features [Schuller et al. 2009] achieved worse results, with a Weighted Average Recall (WAR) of around 60% compared to 40% from [Schuller et al. 2009]. More in-depth analysis of the architecture and a comparable two dimension classification task are done in Section 6.3. This shows that by using visual inputs it is possible to provide different and still efficient features for a machine learning system. While this provides not the same benefits as the CNN system currently favoured, it also requires not as much computational power and less training data before achieving successful results.

## 4.4 Summary of the Chapter

For the research into the first aspect of the concept system, several novel methods for the selection and optimisation of available feature sets were presented. The primary questions of this chapter were, if there are possibilities to improve feature optimisation in complex speech based tasks and if there is an advantage for visual data representations. Considering cases when the ordinary approaches are not sufficient, such as feature spaces with overlapping examples or complex multi-dimensional input arrays, the presented methods provided the necessary improvements to allow the following processing steps in a machine learning architecture to work efficiently. Also discussed were the necessities to reduce the amount of features used, and the reason to even change the internal information representation from an audio signal to an image. Furthermore, the basis for a continuously adapting and engaging recognition system which may optimise its data processing to better align towards a specific user or situation was introduced.

With the optimisation of the used feature sets, both the results of the following recognition steps can be improved, but can also be processed much faster and react potentially in real-time to situations. Building on these effects, the next layer of the system can now more reliably interpret the input. For this different methods will be examined in the next chapter.

Chapter 5

# Processing User Acoustics

## Contents



The second aspect of the conceptualised intelligent assistant system examined in this thesis is the technical ability to "understand" a situation. In practical terms this describes a system which is able to recognise not directly measurable occurrences, such as user states and intentions, environmental states or context depending developments. After I examined the information pipeline from the real world environment into a machine understandable format in the former chapter, it is now necessary for the system to interpret the relevant conclusions from these features. Traditionally, this would be a problem for a technical system when using rule-based approaches, as different feature values would need to be connected to specific states, such as a specific emotion connected to a specific set of voice features [Schuller et al. 2009]. As the complexity and granularity of the extractable real world data increases, manually programmed decision algorithms may lack the necessary versatility or may even be impossible to implement due to the non-linearity of the feature space representation. To solve this problem for complex feature and problem spaces, be it classification or predictive tasks, the applications of autonomous machine learning algorithms are preferable. They can learn the inner context or at least correlations sufficiently to map real world data to conclusions with a satisfactory accuracy, based solely on provided examples.

In this chapter specifically several novel and adaptive approaches for complex machine learning architectures will be examined in greater detail, these can take a functional advantage from the improvements of the features extracted and prepared in Chapter 4. The focus in this chapter lies on architectures combining several functionally independent recognition layers to improve on the abilities of other approaches using only singular functional layers. Further methods concerning architectures motivated by visual inputs will primarily be examined in Chapter 6. The topical structure for this chapter begins first with a short examination of general applications of machine learning methods in my research in Section 5.1. This is followed by two different experimental approaches to layering otherwise single layer methods in a fashion similar to deep learning approaches, but with less computational requirements in Section 5.2 and in Section 5.3. The chapter is then concluded by a summary in Section 5.4, which will also lead to the foundation for Chapter 6.

The relevant basis for this chapter is taken from a variety of experiments done by me, with the primary sources being published in [Weißkirchen & Böck 2018; Egorow et al. 2019] and [Siegert et al. 2021].

## 5.1 Basic Machine Learning

The employment of machine learning systems is based on several advantages. First of all it provides an easy and comparably efficient way to solve a variety of classifying and predicting tasks which would otherwise require complex expert automation designs to solve adequately [Grosan & Abraham 2011]. Related to this is the easy way to implement a human-like training process [Rumelhart et al. 1986], which learns from problems and examples and can improve the results based on them instead of requiring fully explainable expert knowledge. An additional reason is the adaptability and transferability of most solutions from one task to another, which allows the implementation of several solutions based on one basic architecture [Weiss et al. 2016].

The general training process of machine learning requires only the availability of examples, as presented by features such as the ones prepared in Chapter 4, and ideally the prior knowledge what the examples objectively present as labels. Otherwise self-learning approaches are even capable of finding their own conclusions and patterns in situations where no prior knowledge is available and can extract their own features from raw data [Ghahramani 2004]. The method which is most prevalent in this work employs the so called "supervised" approach, where the available features can directly be linked to a specific result in a classifier system.

An alternative implementation would additionally allow such a system to be used as a predictive model [Emmert-Streib et al. 2020], which mainly changes the way the training data is prepared and its classification results are interpreted. The advantage compared to a singular if-then causality connection, is the ability of a machine learning system to designate a region in feature space as belonging to a resulting class, assuring that there exist no undefined situations, even when the situation at hand was never experienced by the system through examples [Sarker 2021b].

A typical example for machine learning systems is the Artificial Neural Network (ANN) architecture, this architecture allows for precisely this kind of training by approximating the kind of connections found in biological neural connections [McCulloch & Pitts 1943]. As further information can be taken from Section 2.2, it is sufficient to say in this place that this original architecture, while adaptable, is often not sufficient for problems beyond a certain complexity and structure [Malinowski et al. 1995]. It nonetheless is a standard building block for a variety of more capable and specialised systems such as Recurrent Neural Network (RNN) [Rumelhart et al. 1986] or Long-Short Term Memory (LSTM) [Hochreiter & Schmidhuber 1997]. Both of these approaches are optimised for time dependencies in the feature space, which is an important aspect of acoustic features in general and speech specifically. This is primarily done by providing a system with its own former results, either by additional connections in RNNs or per special design as in LSTMs. This is the so called recurrence, which may incur often high computational complexity on such problems, as it “remembers” former states of the network for each following step of the computation [Glüge 2013].

An alternative to such methods are more mathematical inspired solutions such as a Support Vector Machine (SVM) [Cortes & Vapnik 1995], which also allows for linear classifications tasks, whose strength lies in the ability to transform high-dimensional feature spaces into more simplistic representations and apply their classifications there. Because of their high robustness and proven capabilities they are often used as a baseline comparison method, either to prove the general classification potential of a corpus or to compare the results against this method [Schuller et al. 2009].

On top of these mentioned methods, which were often used in my research, there exist a wide variety of continuously expanding variants of self-learning architectures [Qiu et al. 2016; Salkuti 2020], with different strengths and weaknesses depending on the kind of available data and the specific type of result which is required [Sarker 2021b]. This shows not only a general trend toward this kind of solution for complex tasks, but also the general versatility a system would achieve by employing these methods in the interpreting layer of the presented assistance system.

Two important problems for most of these systems, beyond the potentially high computational requirement, are the problems of resulting from the “blackbox” behaviour, which describes generally the non-traceability of the found solution compared to a rule-based system. Specifically, this manifests itself on one hand by surprising errors, created even by small changes in the used feature set [Szegedy et al. 2014b], but also in a potential bias based on the used training data on the other hand [Paullada et al. 2021]. Both of these problems, the lack of explainability and the missing impartiality in the system, primarily stem from the incompleteness of the used data and are as such part of the solutions presented in Chapter 4. Additionally, a system may employ a self-check for correctness or plausibility during certain tasks, by using the methods presented in Chapters 7 and 8.

The problem of computational complexity may be solved partially by solutions presented in this chapter, but also by technical solutions such as edge or cloud computing, depending on the type and privacy of the processed data [Murshed et al. 2019; Siegert et al. 2022a]. This would in this case allow the generation, processing and interpretation of the features to be done primarily on the sensor side of the first layer in the information pipeline presented in Section 1.2, while the upper decision making processes done on the central processor would only work on the already processed information from the lower layers.

## 5.2 Deep Machine Learning

One of the main developments in the area of machine learning is currently the employment of Deep Neural Network (DNN) architectures [Pouyanfar et al. 2018; Emmert-Streib et al. 2020; Alzubaidi et al. 2021]. They generally allow for better classification results, even for complex distinction problems, because of their greater generalisation and learning capabilities, the technical reasons behind this are explained on a basic level in Section 2.2. As a rough approximation of their abilities it can be said, that more processing steps layered sequentially are better than one singular processing step [Albornoz et al. 2010; Stuhlsatz et al. 2011]. A common disadvantage is the generally higher data requirement, both to train such a system and to impede the typical effect of overfitting [Ying 2019]. The questions of research in this section are: First, if the proposed method of layering classifiers can be used effectively for tasks using acoustic features, as opposed to visual inputs as usual in Convolutional Neural Network (CNN) solutions? And second, if such a method can be modified to minimise the required amount of training data, instead of increasing the necessary size?

### 5.2.1 Layered Classifier

A possible use of a layered classifier is the implementation of the principal design discussed in Section 4.2. As mentioned there, the ability to distinguish the significant features is of less relevance when they cannot be adequately separated in the feature space. This can either be solved by using a system which can train the complexities behind the feature space and/or by increasing the amount of training data to allow a system to learn an overarching generalisation.

A proposed easy and efficient solution employing this layered architecture would be to first distinguish each source semi-automatically, for example in speaker groups in this specific case, where all speaker share the same characteristics and would fall into the same area of the feature space when deciding the classification of the input [Siegert et al. 2014]. As an easy proof of concept we decided in [Weißkirchen & Böck 2018] to distinguish the speakers based on the biological sex and rough age groupings, as this relevantly changes the way of speaking [Harrington et al. 2007; Siegert et al. 2018]. This was done before applying a typical classification system to the task of identifying high or low mental loads of the speaker. To test the hypothesis we compared, as seen in Section 4.2, the similarity within and outside of this speaker group and found an improved matching inside of the group.

To implement such a system we prepared three different experiments, first of all a baseline where a SVM architecture would be trained on the full available feature set. Secondly, the layered classifier would first solve the comparatively easy task of separating male and female speakers (for which the result by itself is less relevant as the system is only designed to simply group roughly similar sounding speakers together). And as a third approach we employed a Random Forest (RF) as a classifier, as such a system would by design solve a problem by implementing layered decision, this was done to compare the final effectiveness of the proposed system.

The used dataset was the same as presented in Section 4.2 and 3.2.1 and was used for all the experiments in a Leave-One-Speaker-Out (LOSO) procedure. The

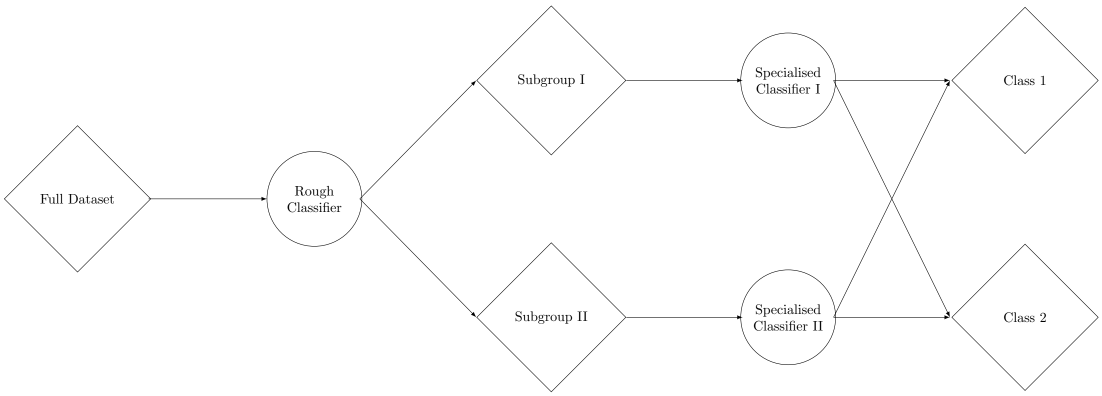



**Figure 5.1:** Figure illustrating the dataflow from the original dataset to the final classification into low and high mental load. The shown configuration used SVMs as classifiers, which all used the same parameters but were trained on different target data. General structure is taken from [Weißkirchen et al. 2018].

layered approach used a dataflow as shown in Figure 5.1 between the different classifiers. The first classifier used the same significant features as the other ones, with the caveat that the significance of the features was not measured for the speaker type classification and as such not optimised for the task. The result was still reasonably high with around 86% Unweighted Average Recall (UAR) classification result concerning the stated sex of the speaker. The results of the experiments can be seen in Table 5.1, which compares the different results for used features and classifiers. As seen, the baseline for SVM is around 68.5% UAR and as comparison 74.2% UAR when using the full dataset with a RF. When using only the significant features, the system first loses its classification capabilities in the classifier approach and regains them only when layering the classification into two steps. These results imply several conclusions: While the difference in recall and precision is comparably small or even slightly negative, the resulting system only uses 20 instead of 933 features. The results for the RF are nearly the same for both approaches, implying that the system is basically following the same principle by design. As a general conclusion one can see the benefit concerning the feature reduction, while at first glance no relevant improvements can be seen for the final classification results. Important for this is the suboptimal training of the classifier in the first step, which would ideally use a different significant feature set for its own task instead of sharing them for both tasks. This would imply that an optimised system would possibly still transfer all available data, while only using a relevant subgroup to reduce computational requirements effectively during

**Table 5.1:** The table lists the Classification Results (Unweighted Average Recall (UAR) and Unweighted Average Precision (UAP)) for all experimental setups. The full feature set consists of all 933 features from the Opensmile's emobase (emobase) set, while the reduced set consist of the chosen ones given in Table 4.1. Full Group and In Group distinguishes if the dataset was separated beforehand for biological sex or not. SVM and RF refer to the used type of classifier. The results are taken from [Weißkirchen et al. 2018].

| | Full Feature Set | | Reduced Feature Set | | |
|---|---|---|---|---|---|
| | SVM | RF | SVM | RF | |
| UAR | 68.5 | 74.2 | 46.4 | 69.2 | Full |
| UAP | 68.7 | 75.9 | 46.5 | 69.6 | Group |
| UAR | 69.5 | 75.5 | 68.4 | 69.8 | In |
| UAP | 69.2 | 75.6 | 60.2 | 69.9 | Group |

each step. Even more complex architectures would be possible, pre-sorting and optimising the full task into smaller and more traceable steps. For this, additional expert knowledge could be implemented like in the presented experiment, or alternatively, the system itself may cluster towards similar occurrences, where even unsupervised methods could be applied.

### 5.2.2 Integrated Feature Optimisation

The next research presented here deals with the idea to improve the feature representation of the available data before training, to ideally optimise the resulting classifier. This is the continuation of the experiment presented in Section 4.3.2, where the effects of the feature selection and reduction process were discussed. The full system, as presented here, constitutes a form of deep architecture, specifically a hybrid architecture as it employs different methods with each having their own set of advantages contributing to the final results. This architecture was used by my colleagues and myself in [Egorow et al. 2019]. An overview is given in Figure 5.2 and the individual steps will be described now.

The first section of the general architecture is the Feedforward Neural Network (FFNN) part of the system, containing the functional sparse autoencoder, which reduces the available data channels for each step of the calculations. In this case, reduction is from the 3.396 channels in the first layer down to 100 channels in the fifth, which is a designated "bottleneck" layer. This means, it presents a reduced information pipeline, where the information has to fit through a lesser amount of channels without losing information. This part of the system has to be pre-trained, before the relevant classifications can be done, as it needs to develop these capabilities of an autoencoder to retain the information of the original data (which consisted of 3.396 different features). This training is done in an unsupervised manner utilising scaled conjugate training methods [Møller 1993], these have to be applied to each individual layer before proceeding to the final bottleneck, which can then be used as an input for another classifier.

The result of this autoencoding in the system is then used for the relevant classification training itself and consists of a SVM employing a radial basis function as kernel, which is only using the processed data from the first part. Compared to typical deep learning architectures, a SVM is much less complex in their computational requirements [Amara Korba & Arbaoui 2018], resulting ideally in less training and/or classification time. The classifier itself was trained independently from the autoencoder. Specifically the autoencoder was trained on a subset consisting only of the female speakers, while the full network was trained with the typical LOSO procedure and the full dataset. This was partially done to measure the potential generalisation capabilities of the system. The general architecture can be seen in Figure 5.2.

As mentioned in Section 4.3.2, the used dataset for the experiment was taken from the Talk Run Speech Database (TalkR) [Truong et al. 2015], which was further distinguished into the subset containing only the 15 female speakers (designated "femTalkR") and the full set with additional 6 male speakers. The autoencoder was only trained on the "femTalkR" part, while the classification was done on the full set (with the mentioned LOSO procedure) to assure that generalisation capabilities can be measured from the classification results. An additional dataset was afterwards used, to measure learned transferable capabilities of the trained system. This was the Munich Biovoice Corpus (MBC), which in contrast consists of 15 male and 4 female speakers [Schuller et al. 2014a] and contains primarily the non-voiced components of the recordings, such as sighs and breathing sounds. Both sets, and the experiment itself, were generated to measure the influence of physical load on the recordable expression of the participants. For these specific experiments only the acoustic aspects were employed.

The experiment tried to answer the following three sub-questions:

1. Can the autoencoder optimise the data representation?
2. Can the effect of the autoencoder freely be transferred between only female speakers and all speakers?

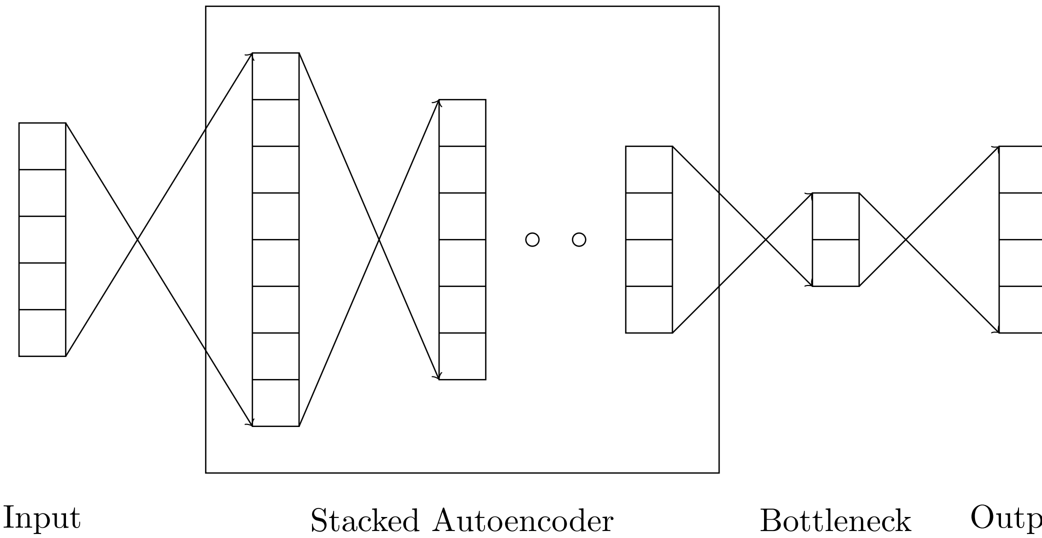



**Figure 5.2:** Sketch of the Bottleneck Feature Architecture. The original input data will be compressed into a smaller "bottlenecked" representation after which the resulting data is fed into a classifier. Not included is the following SVM final classifier step. The concept of the figure is taken from [Egorow et al. 2019].

3. Is the trained classifier generalised enough to transfer to an untrained corpus?

Each step requires greater transferable qualities from the extracted data as the system gradually works on material more dissimilar to the original trained data. Additionally, the available amount of data for this network is much smaller than in typical deep learning architectures, which need normally more examples to achieve generalisation [Amara Korba & Arbaoui 2018]. The result for these experiments can be seen in Table 5.2, it shows that the results for the trained dataset, TalkR, are better than the baseline established in [Truong et al. 2015]. This answers question one and two positively concerning the optimisation aspect. Even though the autoencoder is only trained on the female speaker subset, the effects transfer to the whole set and the resulting classification is better than otherwise possible. The answer for question three is less indicative. It can be seen that the baseline results taken from [Kaya et al. 2014] are better than the achieved results from the system. In contrast to their baseline, the training itself was only done on the TalkR dataset, as such only using general applicable features for the classification. The baseline in contrast trains directly on the correct dataset. As can be seen in this section, the proposed method compared differently to a

**Table 5.2:** The table shows the classification results for the bottleneck optimised SVM classifier. The general relation between the results for female and male speaker is the same as in the baseline, taken from [Truong et al. 2015]. Worse results were seen in comparison with the second MBC dataset, as the system was not directly trained on this data. For this the baseline was taken from [Kaya et al. 2014]. Transferable generalisation results can still be seen. Results taken from [Egorow et al. 2019].

| **Data** | **Females** | **Males** | **Overall** |
|---|---|---|---|
| **TalkR** | | | |
| *- Baseline [Truong et al. 2015]* | *73.5* | *60.0* | *70.1* |
| - Recall High Physical Load | 83.5 | 71.7 | 79.72 |
| - Recall Low Physical Load | 83.3 | 67.4 | 78.73 |
| - UAR Both | 83.4 | 69.7 | 79.22 |
| **MBC** | | | |
| *- Baseline [Kaya et al. 2014]* | - | - | *75.35* |
| - Recall High Physical Load | 56.8 | 64.3 | 61.65 |
| - Recall Low Physical Load | 58.6 | 60.3 | 59.77 |
| - UAR Both | 57.7 | 62.3 | 60.71 |

typical approach. A great boon for the system is the generally low requirement on available training data. With only 250 audio samples (constituting around 85 minutes of continuous recordings), it is far below typical data requirements for deep learning architectures [Dawson et al. 2023]. The extracted features were also comparably easy producible and specifically in this experiment also reducible to a significant extent, from the original 3.396 features down to 100 used data channels in the effective classifying part.

Concerning the required training of the full structure, this process could easily be separated. While the feature reduction and selection process are part of the system, the used autoencoder was, and should in a real application, be pre-trained. With this, the influence on the computational time of the full architecture is negligible. The relevant classifier itself is a SVM with a relative low computational requirement compared to other machine learning architectures, especially deep learning ones. This further improves on the benefits of a small dataset. Together, this gives a positive result for the architecture from an efficiency standpoint, as results are achievable easily and fast.

Concerning the results for the classification process itself, a different conclusion can be drawn. The generated feature representations of the autoencoder were transferable from the purely female speaker subset to the full dataset without a great loss to the achievable results. The following high difference for the classification results themselves appear comparably strong, but they are similar to the baseline results from [Truong et al. 2015], and can be assumed to be an aspect of the dataset itself (based for example on the skewed speaker representation). In sum this provides a good indicator for the system's classification capabilities as a whole.

The final generalisation aspect, its transferability from a trained dataset towards another untrained one, was less successful. As the type and expressiveness of the employed data changed, the extracted representations and classifier proved worse than comparable baseline results [Kaya et al. 2014]. Nonetheless, the system seems to have found indicative results above chance or guessing level of 50%. This proves an adequate applicability potential in this case, even when not sufficient for an optimal result.
To solve this problem a system either needs more informative features generally, and/or better capabilities to employ the available features for a specific situation, with both proposals preferring an individualised approach to the classification task. Given this problem, it leads to put further weight onto the experiments as described in Sections 4.2 and 5.2.1

Concerning the general capabilities of the system, it provides an efficient alternative to other deep learning approaches, as they will be employed in Chapter 6, for the same general setup done comparatively with a CNN system. And, as presented in Section 4.2, it is strongly dependant on the available amount and type of data to decide if such a structure or the more complex approach is the preferred solution.

## 5.3 Continuous Learning

An alternative solution to improve the performance of a machine learning system is by solving the generally appearing problem of lacking training data. While the current trend for assistant systems is to employ great amounts of data [Qiu et al. 2016; Salkuti 2020] this reduces the potential applicability to specific cases, where the designer is capable of providing these requirements. Generally, this "big data" approach attempts to solve a problem by either allowing a system to find the most general classifier for all situations [Kawaguchi et al. 2017] or by providing a sufficient capable system to train its own layered strategies and subtasks to solve a more complex task [Krizhevsky et al. 2017]. As seen in Section

4.2 it is unlikely to find a general set of features and classifier architectures which may solve all tasks satisfactorily, especially in case of highly unique examples sets, as such a true generaliser is unlikely to happen. Also, as seen in Section 2.2, the available systems continuously increase in their complexity, and while the technical capabilities also increase, this often comes with further technical restrictions which are against the widespread adoption, such as exponential increasing processing times. The question in this section is, if there is a different approach which may allow a system to train sufficiently for a complex problem without requiring grand amounts of data before implementation and without computational complex architectures, for a satisfactory result.

For this, one possible solution is a system which would continuously learn from new data, allowing even with small training sets in the beginning to gradually adapt to new information as it appears instead of needing the full training set at the beginning. This would ideally allow such a system to either expand towards new areas in the feature space, which at the moment lack training data covering it, or otherwise to improve specialisation, if the same situation is happening repeatedly with only slight variation in areas where different classes overlap. In case of the voice based classifiers often examined in this thesis, this would translate to a system either learning new speaking patterns or learning the idiosyncrasies of one particular speaker specifically. To examine the capabilities of such a Continuous Learning Framework (CLF) approach, my colleagues and I researched a wake-word independent addressee detector only based on prosodic variations [Siegert et al. 2019].

The used dataset was the Restaurant Booking Corpus (RBC) [Siegert et al. 2019], which is comprised of 30 speakers using phone calls to book a restaurant either via a technical system or a human interlocutor. The full set contains 90 recordings, three for each participant, further information on this dataset can be found in Section 3.2.4. Instead of otherwise typical device-directed and human-directed speech, in that dataset there was no reoccurring wake-word signalling the type of interaction partner. Our assumption as such was that the prosodic variation when knowingly talking with a human is different from the prosodic variation with a technical device, a theory which is based on findings of [Batliner et al. 2008a; Shriberg et al. 2012]. Furthermore, to test the resulting capabilities of the system, we also performed a study with human annotators, especially if they were capable of interpreting the directedness of played samples correctly, also seen in Table 5.3. This setup and problem were chosen specifically, as it is a good approximation for a real world situation where continuous learning could

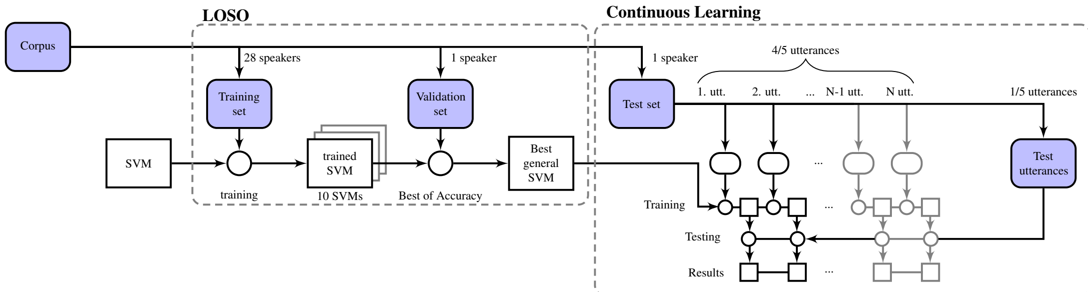



**Figure 5.3:** Presented CLF architecture and dataflow explaining the training and implementation of the principle. By using a LOSO procedure for the pre-training and then employing the remaining utterances of the untrained speaker as incremental interactions with the system, a high similarity to a real use case is achieved. Figure taken from [Siegert et al. 2019].

happen and provide a good alternative to one finalised training approach during system design. Especially during interactions with new users, the ability of a system to rely on old information and examples is not sufficient to provide a correct distinction of the feature space. The architecture for the designed CLF-architecture can be seen in Figure 5.3. An important aspect of the experiment was that the system behaved similar to a real situation, where a new user may interact for the first time with a generally trained system and then gradually starts to adapt in an emulation of continued interaction. As the data was used in a LOSO architecture, the one speaker left out represented this user. The other speakers fulfilled the role of pre-training the system. As can be seen in Figure 5.4 for low number of utterances, the general classification results of the system were considerably low. This is most likely a result of the high individuality of the speaker expressions concerning human and device-directed speech and the general small size of the dataset. It can also be seen in the comparable results in Table 5.3 that the problem by itself is quite complex, even for human annotators or more complex classifying systems. The second step of the training process was then for the utterances of the last speaker to be also separated into individual utterances and then continuously fed to the training mechanism. The approach tries to elicit a certain overfitting behaviour, in this specific case with a positive side-effect as is allows for satisfactory classification results for the relevant speaker where an otherwise general classifier would be of lesser quality. The objectivity was still given as the validation set of the last speaker was separated before the continuous training. Thus the validation was never trained specifically to the system and

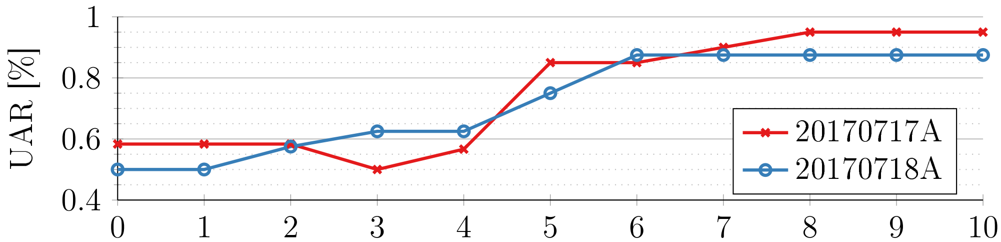



**Figure 5.4:** Progress of the UAR-values utilising our proposed CLF of two speakers from RBC using 10 utterances each. One can see the rapid improvement after around 4 utterances when the system switches to a user adapted state of classification. Figure taken from [Siegert et al. 2019].

only other examples from the same speaker were used for that. Additionally, to improve the training speed the used data was slightly enhanced by repeating the same data with small variations in the features in a form of data-augmentation. Similar to the addition of noise onto training data, this was done to improve the generalisation aspect, while still only employing the interaction from a singular speaker. The experiment has shown that the idea of a CLF framework can

**Table 5.3:** Comparison of average recall and precision values of a two class problem between the human labelers as recognition baseline, our CLF, and an additional meta classifier result from a comparison experiment, taken from [Siegert et al. 2019].

| | UAR [%] | abs. Δ | UAP [%] | abs. Δ |
|---|---|---|---|---|
| Human Labelling (NON-GER) | 53.57 | – | 53.35 | – |
| Baseline (linear SVM) | 52.02 | -1.55 | 52.22 | -1.13 |
| Meta classifier | 62.70 | 9.13 | 63.20 | 9.85 |
| CLF | 85.77 | 32.20 | 85.67 | 32.32 |

be successfully implemented, potentially giving an alternative for the necessary collection of extensive amounts of sample data before the implementation of a system can be tried. The presented system produces very good results, even in this simple exemplary architecture. A potential drawback would be the effect of overfitting, as it may also lead to a decrease of the general capabilities the system had originally, which means decreasing classification results concerning all other speakers. A more adaptive system could use the same overfitting effects specifically to its advantage by, similar to the layered classifier approach in the last section, allowing itself to adapt to different speakers at the same time as being able to generate different classifiers for different situations. On top of that, such a system would also be better for privacy concerns, as the new interaction examples train on the local system itself, instead of sending all examples into external data storage for the generation of future training data as commercially available systems tend to do.

## 5.4 Summary of the Chapter

In this chapter several machine learning architectures were examined, as well as their novel implementations as conducted by me were presented. The main aspect of research was the optimisation of results compared to the usual way of application, without requiring large amounts of training data or computational power. The research here is distinct to the next Chapter 6, where more typical CNNs and DNNs applications will be employed. Regarding this chapter several solutions could be achieved. The general necessity for the use of machine learning methods could be established, especially for the occurring complex classification task in Human-Machine Interaction (HMI) situations. Two effectively novel methods for layering and stacking classifiers were presented, one of them capable of improving generalisation capabilities of a system, while the other reduced the necessary feature size considerably. Finally, the option of using a Continuous Learning Framework (CLF) approach as an alternative to a pre-trained system was evaluated.

In conjunction with the next chapter, this provides the first "intelligent" part of an assistant system, by allowing the classification of internal user states which can be used in the latter decision stages of the system.

CHAPTER 6

# Visual Machine Learning

## Contents



AFTER introducing the general concept of machine learning methods in the last chapter, as well as including potential alternatives for Deep Neural Network (DNN) architectures, this chapter primarily focuses on applying the typical deep architectures in the context of the assistant system. As indicated before, the use of visual classifiers requires different preparations than the ones examined for acoustic data, specifically a change of the format and a generally higher training size requirement. It also tends to require more in-depth parameter adaptions, depending on the relevant problem, to achieve optimal network sizes [Wallingford et al. 2022]. The most important difference in application, which leads to the first topic of this chapter, is the systems own optimisation and adaption towards the usage of visual data representations. In practice it requires data in the form of multi-dimensional arrays instead of one-dimensional number arrays, as is usual in less complex Artificial Neural Network (ANN) solutions presented beforehand to function optimally.

In this chapter, the research is presented first by a general introduction into the Convolutional Neural Network (CNN) deep-learning framework and the effect it has on the typical preparation steps in Section 6.1. This is followed by the practical application in my research for acoustic problems in Sections 6.2.1 and 6.2.2 for different problems. To provide an alternative to the complex CNN architecture, the experiment described in Section 6.3 uses visual keypoints as input values instead of the full image representations as is usual. With this chapter the machine learning examination is concluded in Section 6.4 and leads into the research for better integrations of the human-machine environment with self-adapting and proactively designed methods beginning in Chapter 7.

The primary research used in this chapter is taken mostly from the publications of [Weißkirchen et al. 2017; Egorow et al. 2019] and [Weißkirchen et al. 2020b].

## 6.1 General Visual Classifier

In contrast to Section 4.3, where experimental results where shown on how such a system can be implemented, this section is intended to take a step back to discuss if or when such an approach is sensible for a classification task to pursue. In context of this thesis this has to be viewed as a classification task inside the assistance system pipeline, especially during an ongoing interaction between a user and the system. The initial reason why visual data as basis for machine learning applications, especially for a DNN, is so common is because it follows a general trend in current developments [Nassif et al. 2019; Alzubaidi et al. 2021]. This approach is partially based on the development of the CNN architecture [Krizhevsky et al. 2017], as such a system can employ the locality in the (visual) data representations, which in turn improves the ability of deep learning architectures to correlate even complex patterns through repeating processing steps [Rumelhart et al. 1986]. This is in contrast to most other forms of machine learning, where the data is presented in a vector and all information are processed independently of their relative position to each other. As such the use of deep learning architectures often implies automatically the employment of visual input features themselves, even though alternatives to such an approach exist, such as shown in both external research [Siegert et al. 2019] or my own (Section 5.2) in this thesis. A connected result of this implementation of deep architectures is that it is often in exchange for higher computational and sample requirements [Sarker 2021a] to provide the necessary basis to produce a stable classifier.

The exemplary method to establish this is the mentioned CNN, which employs the convolutional kernel method, explained in Section 2.2.1, to establish correlations between "regions" of information in each processing step. With these an area of input is processed as a whole, where each additional layer of the system increases the area of processed patterns from the original input as well as establishing an internal data processing pipeline to optimise the flow of information.

This allows the system to generate its own process of data interpretation similar to methods otherwise used in pre-processing done for other architectures, such as resembling typical visual filters when used on images, as shown in [Krizhevsky et al. 2017], where an edge filter trained itself to interpret the visual input. Based on this architecture as groundwork, a wide variety of adaptions developed, which concentrate on different aspects of the method, for example on different depths when employing functional layers, on better visualisation of the data flow inside the processing steps or even just small adaptions on how data is processed inside the kernels itself [Pouyanfar et al. 2018; Sarker 2021a].

A general alternative and precursor for the use of CNNs is given by the already mentioned architectures in Section 2.2, which includes the Recurrent Neural Network (RNN) and the layered architecture as mentioned in Section 5.2. Specifically the recurrent networks achieve their "deep" characteristics by processing the information several times through the same layer for each time step, instead of the otherwise used method of layering the functional units behind each other [Rumelhart et al. 1986], which reduces the complexity of implementation but also reduces the possible extent on how complex the solution can be.

The cost of the comparably complex architectures is the increased requirement of training samples to achieve generalisation effects during training. An otherwise frequently observed effect is the overfitting to the available training samples, which inhibits the further use of the system when new data needs to be classified. As an example, when using Leave-One-Speaker-Out (LOSO), such as in the experiments done in [Weißkirchen et al. 2017], one observable effect was a fast increase in classification accuracy for the training set, achieving nearly perfect recall and precision during training, while also remaining on a practical random chance level for the validation set, which comprised of utterances from speakers which were not trained during the creation process. The same effect could be observed even when the validation sets consisted of untrained utterances coming from the same speaker which also provided the training data. This strong effect could only be reduced by extensive data augmentation and expansion, after which the generalisation capabilities of the architecture began to set in.

These high requirements, in conjunction with the otherwise high capabilities of the system, prove a strong indication for the possible use cases, especially when aiming for widespread applications, where each user is required to participate in the data generation, as is usual for current assistant systems. In these cases it can ideally work as a complex generalising classifier, as it can process and train continuously to include new information. Additionally, it remains a very capable implementation for its originally designed image classification task, where the general data pre-processing methods are trained on big data sets and the further optimisation to a specific task can be done on smaller sets by the user himself [Krizhevsky et al. 2017]. The applicability of successfully trained architectures for one task when used for another classification problem is strongly reduced when different forms of data representations are used, which cannot employ the same visual filters to a similar effect. This requires often a more in-depth retraining of the underlying CNN architecture.

In specific cases the necessity of DNN for these positive effects is not given, such as in the examples of Section 5.2. This approach to employ alternative methods for the same effect will be examined further in this chapter, especially in cases when these alternatives are preferable, see Section 6.2.2.

## 6.2 Convolutional Classifier

In this section the research done concerned one of the usual main DNN methods, which is the CNN architecture. At the time of my first experiments, this architecture was not typically used for acoustic input. This fact necessitated a more thorough examination of the parameters, which is presented in Section 6.2.1, on how the system had to be adapted for this unusual data type. In Section 6.2.2, a comparable experiment to the one presented in Section 5.2.2, is examined with an alternative architecture, the results are then used to establish a baseline for the decision on which type of architecture is preferable for different classification tasks, based on external factors.

### 6.2.1 Visual Classifier for Acoustic Features

Even though the application of CNN architectures provides the common state-of-the-art of deep learning methods [Krizhevsky et al. 2017], it is still primarily based on the interpretation of visual data. Further explanations about the structure and development of the architecture can be seen in Section 2.2.1. As a short reminder, the original architecture is based on the visual cortex of most mammal species [Krizhevsky et al. 2017], as such, both the processing as well as the input format are solely structured around this specific application case. Conversely, it provides great processing improvements in the areas of classification results and complexity for these specific cases. As a result, research by me and my colleagues from 2016 to 2017, for example in [Weißkirchen et al. 2017], was done to transfer these capabilities into the acoustic area of speech interpretation, such as emotion classification from spoken words. To our knowledge this was one of the first forays into this area, at least with published results, and was done to explore the general possibilities and potential improvements which may come with this change of frame.

When adapting the otherwise used methods to this new problem space, it was first necessary to employ a transfer from the usual acoustic representation, similar to the pipeline explained in Section 1.2. Several options appeared possible at the time where no best practice was established yet.

The typical approach employed was the use of feature extraction methods, already proven in the field of speech interpretation, such as the Opensmile's emobase (emobase) feature set [Eyben et al. 2010]. This method, which is usually employed for ANN, gives all features in a one-dimensional array for the system to process. The alternative to this, which is inspired by the specific processing steps in the convolutional layers explained in Section 2.2.1 is specialised on working on two or more dimensional inputs, is the transfer of the acoustic data into the visual space. The easiest and fastest of the used transformations was a simple wavelet representation of the utterance [Debnath & Antoine 2003], which in the experiment proved insufficient, most likely because of the sparseness of information compared to the size of the image. The more useful and chosen alternative was the more complex spectrogram representation [French & Handy 2007], as seen in Figure 6.1. Practically the aim for the data transformation was to improve on how much information is coded in the available channels of an image, supporting the convolutional steps of the CNN. Colour, depth and time can all be used in this approach to improve the amount of information available, for which the spectrogram was chosen as a representation, being both rich in information density and still be easy for a human observer to understand. The usual approach for a new classification task with CNN architectures is the employment of a pre-trained network, where only the final classification layer, usually composed of several simple fully-connected feed-forward steps, will be changed. This allows the pre-trained visual identifiers, as shown in [Krizhevsky et al. 2017] resembling edge and point detectors, to transfer into the new problem, practically only changing the classified object from one class to another. This optimisation for (physical) object detection was not adequate for any of the typical used representations of acoustic data, therefore the research further concentrated on the full new initialisation training of the network and the parameter optimisation

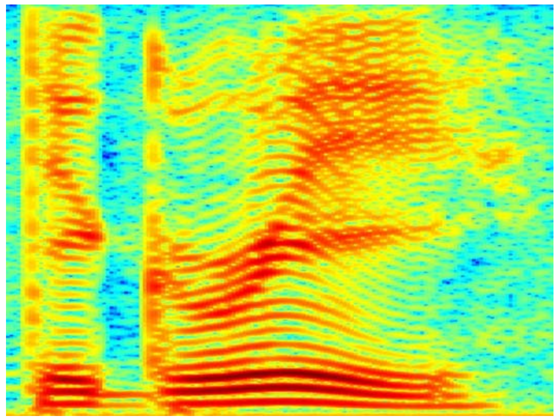


**Figure 6.1:** A colour Spectrogram of the word "Degree" from the Speech Under Simulated and Actual Stress (SUSAS) dataset. As can be seen, the different parts of the image all contain important information. Frequency and time is part of the x- and y- coordinates, while the amplitude is coded in the colour channels (3 for the typical rgb approach). A simpler version could use grayscale information and would then only require one channel. Taken from [Weißkirchen et al. 2017].

combined with that. As a result of the new training, I employed a comparatively small depth for the architecture, similar to the one used first in [Krizhevsky et al. 2017]. Additionally, the use of a relatively small data set employed in the research, required further restrictions on the resulting network. The used dataset was the Berlin Emotional Speech Database (EmoDB)-corpus, consisting of ten actors speaking 553 phrases, specifically in a form which contains acted emotions to inflect on each of the utterances. The speciality of this dataset is that the emotional inflections are quite distinct and as such the different classes differ clearly from one other. This often leads to the employment of this corpus as a baseline method [Schuller et al. 2009]. Further information of this dataset can be taken from Section 3.1.1, specifically how it was originally designed as base for inflection experiments. Furthermore the eNTERFACE'05 (eNTERFACE) corpus and the SUSAS corpus were employed to provide further comparisons with [Schuller et al. 2009]. The eNTERFACE corpus consisted of 1.277 examples, the SUSAS corpus of 3.593 examples. The setup of the architecture itself was changed dynamically during the experiment by employing a LOSO validation set. As mentioned, the optimal structure was different from the otherwise employed architecture of the original visual experiment. Both complexity and available data sizes were smaller, as such the structure itself was reduced, as seen in Table 6.1. It was not necessary to employ the same amounts of kernels, or convolutions, and the amount of

**Table 6.1:** Structure plan of the designed CNN system, the functional aspects consist primarily of convolutional layers. Additionally, the system employs a "pooling layer" which practically reduces the processable amount of information by pooling several kernels together. The later layer follows a more traditional Feedforward Neural Network (FFNN), consisting of a neural net with Rectified Linear Unit (ReLU) as activation functions and dropout layers to reduce overfitting. Taken from [Weißkirchen et al. 2017].

| Layer Name | Filter Size | Step Size | Depth |
|---|---|---|---|
| Input Layer | | | 3 |
| Conv 1<br>Max Pooling | 11x11<br>2x2 | 5<br>2 | 96 |
| Conv 2 | 5x5 | 1 | 256 |
| Conv 3 | 3x3 | 1 | 384 |
| Conv 4 | 3x3 | 1 | 384 |
| Conv 5<br>Max Pooling | 3x3<br>2x2 | 1<br>2 | 256 |
| ReLU<br>Dropout Layer | 4096 | | |
| ReLU<br>Dropout Layer | 4096 | | |
| Output Layer<br>Softmax | Variable | | |

**Table 6.2:** Unweighted Average Recall (UAR) of the classification experiment of different datasets, top1-top3 represent when the correct class is part of the highest probabilities taken from the top 1 to the top 3. Development Set consists only of untrained LOSO examples. Taken from [Weißkirchen et al. 2017].

| | Avg. UAR on Test-set | | | Avg. UAR on development-set |
|---|---|---|---|---|
| | top1 | top2 | top3 | top1 |
| EmoDB | 0.71 | 0.86 | 0.94 | 0.96 |
| eNTERFACE | 0.66 | 0.73 | 0.86 | 0.87 |
| SUSAS | 0.57 | 0.76 | 0.89 | 0.92 |

layers was also reduced in turn to suppress the overfitting effect. Additionally, a stronger dropout was implemented to achieve better generalisation effects with higher robustness. With the data generation methods, also mentioned in Section 2.1, the available data was expounded by data augmentation methods described in Section 2.1.2. The final results, as seen in Table 6.2, were still relatively low. But when including the second and third highest probabilities of the results generated, it achieved nearly the same high results as the ones from the training set. While this would normally be no indication for success, the difference in probability for the most likely classes only manifested in the lower percentiles of distinction, which was then drastically increased for the following classes. An explanation for this non-optimal result with ambiguous class distinction can directly be linked to the overarching overfitting effect which even with all the used methods could not entirely be eliminated. As this experiment was done at a time when no, or very few, published methods existed to adapt the architecture to a small dataset, no better results were achieved at that time. Still the method proved positive indications, especially as the result itself were comparable for the dataset to the ones used as baseline in [Schuller et al. 2009].

With this experiment a first indicator for the employability of this DNN were established. Data size and parameters in turn proved to be very important deciding factors for the use of CNNs, as it also indicated that a system which receives a sufficient amount of examples, would be capable of providing much better results than otherwise employed machine learning solutions. In general this experiment showed a promising starting point for the employment of these architectures, and for the classification capabilities for non-visual data.

### 6.2.2 Convolutional Feature Optimisation

With the more widespread adoption of the CNN architectures, even beyond the original visual representation frame, the generally positive effects of this method compared to other machine learning architectures lead to certain expectations of general best practice. As indicated in Sections 2.2.1 and 5.2.1, this may not be always the case. The overall assumed positive abilities, such as good generalisation capabilities and generally higher classification results than other machine learning methods, have to be weighed against the higher resource requirements of the system, be it computational or of the available example sizes [Sevilla et al. 2022]. This section specifically examines the question if and how a CNN architecture can be optimised or adapted for different applications, with a special emphasis on the amount of available data and how it compares to similar approaches. To get a basic comparison of applicability, my colleagues and I performed such an examination between two approaches in [Egorow et al. 2019], one of them being the aforementioned CNN while the other was a more traditional FFNN approach.

An added benefit which has to be acknowledged is that the use of CNN can potentially remove any further pre-processing steps, except providing the data in a readable format for the system. Specifically this means that slight optimisations for different areas of the input are done by the system itself, instead of manually by the supervisor before the training [Krizhevsky et al. 2017]. As such CNNs are capable of performing feature selection similar to the methods explained in Section 5.2.2, where an autoencoder was used to optimise the data representation. A disadvantage of the convolutional steps during this process is the further increase of data requirements which needs to be computed instead of a preferable reduction. This is a result from the architecture in which each input layer gets processed with several independent filters. Admittedly this also depends on the number of used and necessary kernels, as explained in-depth in Section 2.2.1.

The experiment done here is in comparison to the one done in Section 5.2.2, while there the usage of an autoencoder was exemplified as a method to lead information from the input into a classifier. In this experiment instead a CNN is used, this means that instead of separate classifiers and input architectures, in this experiment all processing steps were done in one architecture. The underlying structure is nonetheless similar, as the structure of the typical CNN is also separated into two aspects. First are the convolutional layers which process the input (images) into an efficient format for the data, without losing information. Following this are the classifying steps which are done by employing a relatively simple feed-forward neural network. This allows basically for feature input optimisation and classification optimisation separately, especially after the underlying problem is trained. The otherwise used data and setup is similar

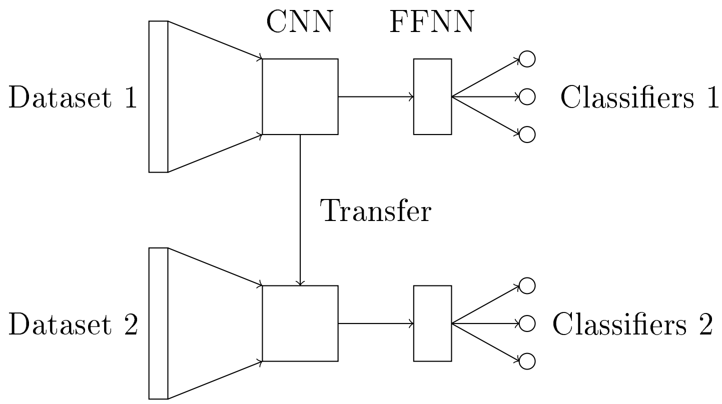



**Figure 6.2:** Seen is the concept to transfer CNN networks from one system to another to significantly reduce training times. For this to function at its best, the different datasets need to be similar in their expression. Only the final interpreting layer (often a FFNN) needs to be trained on the new possible classes.

to the comparison experiment, also employing the distinction between the Talk Run Speech Database (TalkR) and Munich Biovoice Corpus (MBC) to observe the transferability of the different experiments, practically measuring the generalisation capabilities of the system itself and follows the build in Figure 6.2. It also compares the general effort in setting up the system. Compared to the experiment done in the last section, the application of an acoustic input and the implementation of the architecture itself were more advanced, as follows: The data representation also employs a spectrogram, similar to the last section. Instead of using colour as an encoding for volume, here a black-and-white image was chosen, which only uses one channel for the grayscale values. The other channels contained the first and the second derivative of the spectrogram, enriching the information of the chosen data representation. When examining the results from the experiments, as seen in Table 6.3, it appears a bit different from the one done before. One can see, that again the system works reasonably well to generalise between the female part of TalkR and the full dataset. Also similar to the previous experiment and the baseline is the ability to classify correctly slight differences between the male and female speakers. Here the generalisation is working as intended, without the strong overfitting effect seen in the last section. Contrary to this, the transfer from the TalkR dataset to the MBC corpus is

**Table 6.3:** Classification Results from the classifier generalisation experiment. Generalisation on the same set where the training data was taken from is similar to the baseline, while the transfer of the classifier to a new corpus reduced the results considerably. UAR stands for Unweighted Average Recall. Table values taken from [Egorow et al. 2019].

| **Data** | **Females** | **Males** | **Overall** |
|---|---|---|---|
| *TalkR* | | | |
| *- Baseline* | *73.5* | *60.0* | *70.1* |
| - Recall High Physical Load | 81.7 (83.5) | 74.3 (71.7) | 79.94 (79.72) |
| - Recall Low Physical Load | 82.0 (83.3) | 72.5 (67.4) | 79.09 (78.73) |
| - UAR Both | 81.8 (83.4) | 73.2 (69.7) | 79.52 (79.22) |
| *MBC* | | | |
| *- Baseline* | - | - | *75.35* |
| - Recall High Physical Load | 56.9 (56.8) | 55.9 (64.3) | 56.20 (61.65) |
| - Recall Low Physical Load | 55.0 (58.6) | 55.3 (60.3) | 55.19 (59.77) |
| - UAR Both | 55.9 (55.9) | 55.5 (55.5) | 55.69 (55.69) |

considerably worse. Here the mentioned overfitting effects come into play, exemplarily by only achieving slightly better results than chance level, clearly below the comparison and baseline experiments.

These aspects of the result, together with the implementation itself shows the capabilities but also the disadvantages of the system. It employs the efficient data extraction without further human effort, given the optimised presentation of the input data itself as an image. As shown, the overfitting also can be reduced to a smaller effect as long as the data itself is relatively similar, as is the case for data from the same corpus. When transferring the same classifier to another corpus the effect becomes apparent again. This reduces the applicability of the architecture, especially in real world applications. When now comparing both approaches directly, under the assumption of application in an assistant system, one can see the area of application depending on the available amount of training data and the type of classifiable data. The current trend is by employing great amounts of data, preferably from a multitude of sources ensuring a generalisation over different types of data. When assuring that the type of data remains relatively the same, as seen when using the same dataset even when not using male speakers, available pre-optimisation such as data representation and augmentation suffice. With this also smaller datasets can be used as training base.

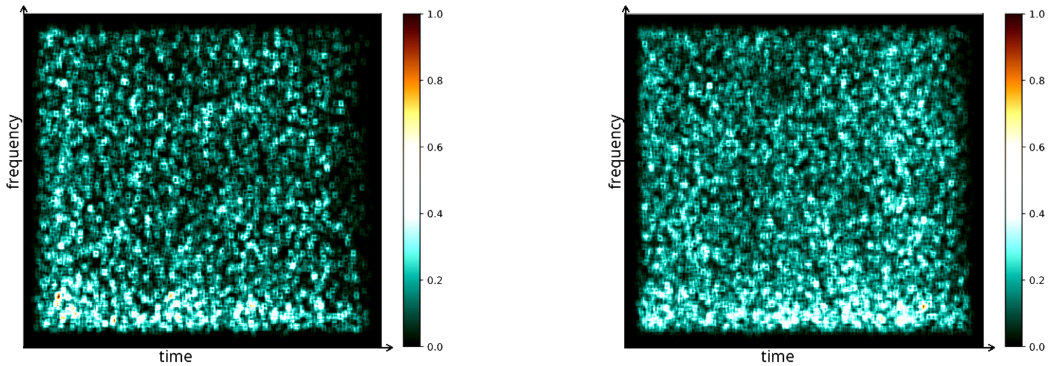



**Figure 6.3:** Marked Keypoints for a neutrally intoned Word and one spoken in an angry voice taken from [Weißkirchen et al. 2020b]. One can see that keypoints cluster in different areas of the visualisation, instead of using the full image as input only the parts containing relevant information need to be observed.

## 6.3 Visual Feature Classification

An altogether different method of interpreting visual data, instead of using the aforementioned CNN, is by employing otherwise typical image processing steps [Rublee et al. 2011] and feed the resulting information into a machine learning application. This functions as an alternative variant to the typical approach for deep learning architectures, and can also be described as a hybrid architecture. In this experiment, as was already introduced in Section 4.3.2, spectrograms were also used as input, given their high density of information over the full image representation, and the assumed existence of repeating patterns for the same class [Weißkirchen et al. 2020b]. The specific difference to the otherwise used method is that the feature extraction itself searches for repeating patterns, or keypoints, which then were used as input for a Support Vector Machine (SVM) as classifier. This offers an alternative to the idea of using CNN, with all its potential drawbacks, while at the same time using the advantages of the visual representation with its high information density and additional locality aspect. As shown in the last section, the applicability of deep learning architectures can be highly dependent on the specific situation, with the availability of data being especially important. By using a less complex classifier, it was aimed to reduce this dependency on the dataset size.

The used classifier consists of two different functional aspects. The first aspect employs an image processing method or computer vision algorithm, called Oriented FAST and rotated BRIEF (ORB), to identify repeating patterns [Rublee et al. 2011]. The second aspect is then a regular SVM classifier, using the existence of the pre-trained keypoints as information for the classification step. As

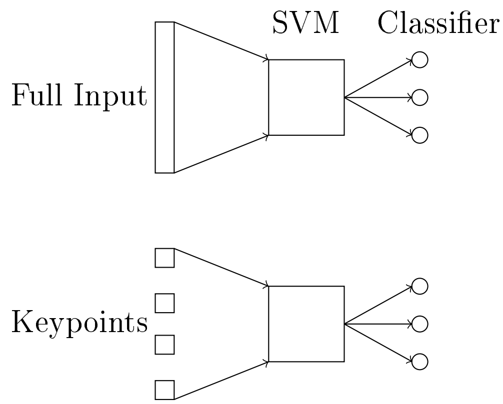



**Figure 6.4:** Seen here is the optimisation of a Classification Network by using keypoints. Instead of training a network to interpret a full visualisation of the data, it only needs to employ the much sparser, but information rich, keypoints. The network in this case can be less complex and trains much faster without losing recall or accuracy.

shown in Section 2.2 the computational requirement for a SVM are much lower than for a DNN or specifically for a CNN. Additionally, they do not tend to overfit as easily with few examples [Sakr et al. 2016]. As such the system ideally benefits from the improvements of the visual representation without increasing the requirements at the same time. As an added part of the experiment we also tried to reduce the necessary features for a functional classifier, further reducing the minimal computational requirements, the experiments is shown in Figure 6.4.

The used dataset for this experiment, is the SmartKom Database (SmartKom) [Steininger et al. 2002], which is also further explained in Section 3.2.3. As a short summary, the dataset contains 3.823 natural acoustic emotion samples, classed according to a variant of a typical emotional chart, containing: anger, helplessness, joy, surprise, neutral, pondering and anger. As an alternative we also employed a classification only using arousal and valence as output, effectively lowering the dimension of the classification from seven to two [Russell 1980]. The system proved functional and advantageous compared to the chosen baseline experiment in [Schuller et al. 2009]. Even with only 10 keypoints the system achieved results of 60% Weighted Average Recall (WAR) compared to 40% taken from the baseline. The weighting was done to account for the high numerical

**Table 6.4:** Results for the classification of Valence and Arousal in comparison to the baseline taken from [Schuller et al. 2009], the experimental results are taken from [Weißkirchen et al. 2020b]. Even in the reduced problem space the chosen method provided significant improvements in the classification.

| Class | Base | Results | Diff. |
|---|---|---|---|
| Arousal | 64.1 | 82.4 | 18.3 |
| Valence | 75.6 | 91.3 | 15.7 |

occurrence difference between the different classes. When changing the classification task to a valence/arousal classifcation, the results also improved compared to the baseline, as seen in Table 6.4. Interestingly, the results of the system relating to accuracy and recall decrease when employing more keypoints. This either implies that the problem became too complex for the system, or more likely that with the inclusion of too many keypoints there occur overlapping information in the problem space which can be counterintuitive for the classifier. Specifically in this case this could be the similarity in utterances, chosen words or other aspects which are not directly connected to the emotion classification.

## 6.4 Summary of the Chapter

In this chapter different examples for the usage for visual representation of acoustic signals were examined, specifically in context of machine learning methods and their applicability for classification tasks. The main research was in their employment for deep learning architectures, which were primarily used in the form of CNN architectures. This was chosen as an example for an approach which is based on the present state-of-the-art for a complex machine learning system. Alternatively, a method for employing the visual input with a less complex architecture was also presented here. This variant allowed for the higher information density contained in an image compared to a value array to be interpreted efficiently, without also requiring the high increase in computational power. A comparison of all the results discussed can be seen in Figure 6.5. In conjunction with the former chapter, this concludes the purely machine learning aspect of the proposed assistant system, which is functionally the second layer of the full system. In the following chapters the transfer from a purely passive system towards a more reactive and finally proactive system will be explored. These additional experiments also employ machine learning processes as a tool, but concentrate the examination on the further improvements done to provide the necessary basis for the later development towards a peer-like system.

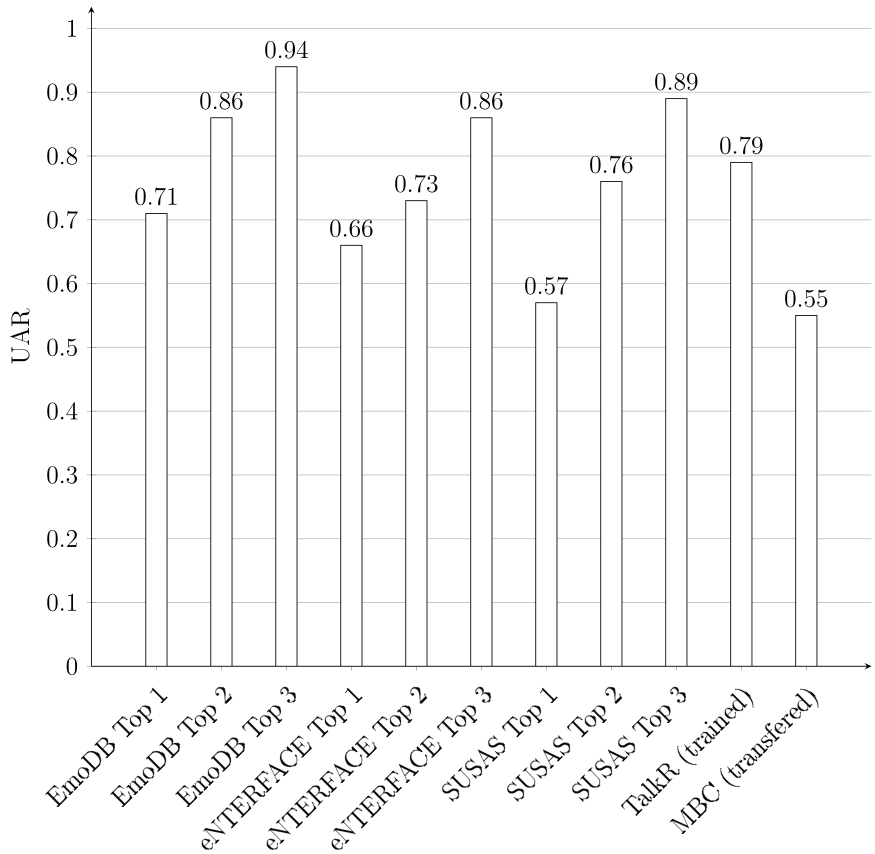



**Figure 6.5:** A comparison of all the results given in this chapter, as can be seen all results are above chance level, most pronouncedly so.

Chapter 7

# Application of User States

## Contents



In this chapter, the usable results and possible implementations of the formerly investigated machine learning techniques will be examined in greater detail. While the last two chapters established the general capabilities and advantages of machine learning over purely rule-based approaches, it did not establish its concrete use in the frame of an assistant system which can show empathy to its user. For this, the most prevalent implementation would be the classification of inner user states and implicit situational information.

As mentioned before, a personal assistance system is ideally operating in an integrated human-machine environment, which requires more information from their users than a simple speech-to-text parsing with word detection of the voiced input can provide. While humans have a natural ability to interpret information which is not directly stated, machines lack the inner ability and understanding of these cues.

With sufficiently observed correlating occurrences of measurable features, taken from exemplary interactions, and at the same time the expression of the not directly measurable inner user states, taken ideally from the users through self-reporting, the system can train a practical approximation of this natural ability in a technical form, even without requiring a direct causal connection between the features and the states. As an example, different relevant user states are described in this chapter, with their impact in the full theoretical framework explained. In Section 7.1 an introductory overview of the type and effect of user states in an assistant system are given. This is followed by Section 7.2, which proceeds with specific examples of emotions in Section 7.2.1, the mental load in Section 7.2.2 and physical load in Section 7.2.3 as practical examples of inner user states. In Section 7.3 another example of implicit information detection is shown with addressee-detection, this information is not part of the inner user states usually associated with paralingual features and proves further potential of this application. In Section 7.4 this chapter is summarised. An important point to mention is the repetition of the experimental results of prior chapters, while the specific setups and methods will not. The main aspect here is an observation and discussion of these results in the frame of the projected assistance system, specifically under the constraint of minimal privacy invasion by using only or mainly audio cues, instead of video recordings or wearable sensors.

As this chapter is a collection of different aspects, the examples are taken from a wide variety of publications, mainly from [Weißkirchen et al. 2017; Weißkirchen et al. 2018; Weißkirchen & Böck 2018; Egorow et al. 2019] and [Weißkirchen et al. 2020b].

## 7.1 Relevance of User States

The current state-of-the-art in commercially available assistant systems, including their variations such as smart homes and smart factories [Lee 2015; Rock et al. 2022], primarily concentrates on the ease of control, specifically using speech as a medium [Rakotomalala et al. 2021]. Based on this trend, the implementation of the voice controlled interface continuously improves in its capabilities, often representing the main aspect of human-machine interaction and the stated aim for future developments. In contrast, or at least as an extension, of this idea is the aim of a technical companion [Biundo & Wendemuth 2017], with assistant systems in particular being a part of that, where the aspect of understanding and engaging the human user in a personalised and situational dependent manner is of much greater importance than being just a user interface. These systems should ideally adapt to their user and their individual abilities instead of providing a general assistance to most users.

Contrary to true human-to-human interactions, technical systems by themselves lack the ability to empathically interact, as they lack the behavioural basis for this [Plutchik 2001]. Additionally, even the approximation of this behaviour with rule-based solutions requires further information, which the typical voice interaction does not convey [Campbell 2004], which requires different approaches such as the development of systems capable of recognising affective states of the user [Picard et al. 2001]. In this chapter the specific aspect of lacking background information is the focus, specifically which information of the internal user state can be extracted, without requiring further specific instructions from the interaction partner. This also excludes information generation through other methods, such as further external sensors or additional statements which may intrude on the user behaviour or lifestyle for the sake of such a system [Chignell et al. 1999].

The first step of enabling a system with an approximation of human empathy, is by converting the user states into specific categories, as described in Section 4.1 where natural signals were digitalised. As described before, psychological expressions such as emotions, physical- or mental-load are not directly measurable values, in contrast to temperature or voltage, it has to be determined indirectly by identifying correlating physical features such as taken from biosensors. This also provides an exemplary application of the methods presented in the last two chapters, Chapter 5 and Chapter 6.

Such a solution, for generating indirect indicators, for the internal user states can theoretically be applied to a nearly limitless amount of different states, as long as the found relations are reproducible in similar experiments or situations. In this chapter, three specific applicable use cases will be examined in greater detail: This will be the emotional state, the physical load and the mental load of a user. It is important to reiterate that such a system does not possess a direct understanding of, for example, the emotion sadness or happiness and how to react in a natural way to it. It instead requires a follow-up rule, as in rule-based system, on how such a measured emotion may change the following decisions of the system itself and how it can react ideally. As such it presents still a fully reactive, technical approach of human behaviour.

Furthermore, general information from the user may also be described as user state, this area can include more in-depth descriptions such as affect or interest of the user and may require the successful interpretation of facial expressions, gestures, or body posture in the general classification step. In its completeness such information can be used to mirror the inner processes of a human in a technical understandable format [Picard et al. 2001]. Even though such a system still lacks true human empathy and understanding of these states, it can allow for a functional sufficient approximation, which can be used by the technical system for its decision making process.

As my research approaches a technical assistance framework from the point of view of a human-like partner and supporter, it in turn needs relevant information to achieve a approximation of empathy for its user. Either by using the rule generating framework presented in later chapters, or simply as a pre-defined causal connection between a certain user state and a specific assistive action. Exemplary results for these states will be presented in the following sections, with a view on how they can be integrated into the wider architecture.

## 7.2 Inner User States

One of the major aspects in my research concerned the recognition of the inner user states of human speakers [Weißkirchen et al. 2017; Weißkirchen et al. 2018; Weißkirchen & Böck 2018]. The biggest topic in this regard was the area of human emotions [Weißkirchen et al. 2017; Weißkirchen & Böck 2018]. This includes primarily a general emotional state of the users, such as their happiness, sadness or similar broad categories. These distinguish the subjective expression of the users concerning their current situation, which can then be used to change the approach the system may employ towards them. In addition to this general expression, the users may also exhibit specific states concerning their current mental or physical load [Weißkirchen et al. 2018; Egorow et al. 2019], with the general term of load in this case meaning a measure for the specific impact of a task on an individual user. While the perception of load is highly subjective from the user's part, the indirect measurement of this perception allows for awareness of the system, when the user is stressed [Weißkirchen et al. 2018].

### 7.2.1 Emotions as User States

The first area approached for the inner human states is the aspect of emotions and how they can be distinguished into classes. This was partially described in Section 2.2.2 in its basic form as part of machine learned classification tasks. In this section instead the potential capabilities of this approach and their practical implementation in the assistance system in the frame of the research I have done will be discussed. While the human awareness of emotion is often additionally dependent on a variety of features, such as facial expressions, gestures, as well as depending on the situation itself, a purely voice based recognition is possible, and technically implementable [Ververidis & Kotropoulos 2006; Sezgin et al. 2012], which leads to the question if this is a realistic way of approaching an artificial sense of empathy for a technical system, given the wide variety of external situations such a system may require to cope with.

As the basis for most of the research into this aspect, it is necessary to define the natural emotional expression into discrete recognisable classes [Batliner et al. 2003]. For this, different approaches are possible and were also used in my experiments, such as in [Weißkirchen et al. 2020b], where different possible classification schemes for the same dataset were presented. A common scheme is the use of around seven classes based on the work in [Schuller et al. 2010]. As this distinct class representation is comparably unnatural for a human like interaction, two different improvement methods are usually employed, either by increasing the number of distinctions, or alternatively by reducing the number of classes into more general representation as in [Schuller et al. 2010]. The first approach requires a greater number of examples which are evenly distributed for these classes and the necessary annotators for preparing, and was not used in my experiments. The other method assumes a continuous emotion “space” which is described by its two axes of valence and arousal, which is based on a different approach to observe emotions [Wöllmer et al. 2008]. Comparative research allows mapping all other emotion classes into this representation. This was used for example in [Weißkirchen et al. 2020b].

To receive emotional data, such as in the datasets mentioned in Section 3, these emotions had to be either acted or induced. Natural emotional states are comparably hard to come by for a data collection, especially when it needs to be equally distributed and unambiguously annotated, requiring the artificial generation of these examples instead, either through acting or inducing the emotion by or in a speaker, alternatively by using data augmentation methods to inflate the available amount of examples. During acting, such as in the Berlin Emotional Speech Database (EmoDB) corpus, the user strongly exaggerates the expression compared to natural occurrences, while induced emotions in contrast have the problem, that some emotions are harder to generate than others.

The reason for this research is that emotions represent one of the main paralinguistic features employed during interpersonal interactions, which allows for a wide range of interpretation about the personal state of the speaker without inhib-

**Table 7.1:** The table shows the development from 2009/2010 to the ones done in my research for different datasets. Further information of the results can be taken from Chapters 5 and 6 where the experiments were presented. The baseline is taken from [Schuller et al. 2010] from the Support Vector Machine (SVM) experiments. Depending on experiment the Unweighted Average Recall (UAR) or Weighted Average Recall (WAR) results were taken.

| Corpus | Source | Result |
|---|---|---|
| emoDB(Baseline) | [Schuller et al. 2010] | 0.86 [UAR] |
| emoDB | [Weißkirchen et al. 2017] | 0.70 (0.96) [UAR] |
| Smartkom(Baseline) | [Schuller et al. 2010] | 0.39 [WAR] |
| Smartkom | [Weißkirchen et al. 2020b] | 0.60 [WAR] |

iting the transmission of textual information through the speech itself [Campbell 2004]. For a technical system to assume the same capabilities during a Human-Machine Interaction (HMI) as is usual between human speakers, a system needs to be able to record and interpret these same signals, or at least include them in the interpretation of the spoken words, as the emotional state may strongly influence the meaning of a sentence [Ephratt 2011].

This usage in context of an integrated assistant system is as such manifold, as emotional user states can be used in that case to indicate satisfaction, interest or agreeableness during all interactions. Potential errors can, for example, be assumed by continuous aggressiveness on side of the user leading to the system to react accordingly. The broader distinction into user affects improves on these indicative abilities by including further reaction states to account for non-emotional reactions. My research followed a basic approach of utterances based emotion detection, usually employing the Opensmile's emobase (emobase) feature extraction pipeline [Eyben et al. 2010] with different classification methods, such as SVM and Convolutional Neural Network (CNN), depending on the available amount of data and the complexity of the used dataset. For more in-depth explanations of the experiments itself, they are described in Chapters 5 and 6 as they also represent a form of machine learning architectures. As can also be seen from the Table 7.1, the results were clear improvements over the ones from several years back. Additionally to the direct work done by me and my colleagues, the area of emotion recognition is generally on a trend of improving accuracy and applicability [Siegert et al. 2022b]. This shows not only that the use of human emotions in technical system is quite possible, but that it is also still improving, potentially even above the original human level of understanding [Siegert et al. 2021].

For the resulting technical system itself, this can lead to several improvements, as otherwise overlooked paralingual features can then be used to improve understanding and reactivity from the technical side towards the human speaker. It can also be used as an alternative form of information exchange by itself, without requiring the direct expression from the user and instead changes in emotion or state may prompt a technical system to initiate certain actions. Concerning the described assistance system it can be concluded, that emotion recognition, even when only using voice, is a stable and important aspect of HMI.

### 7.2.2 Mental Load as User State

The second aspect of inner human user states examined here is the area of mental load. This term is relatively widely applied for different topics, such as in [Liz Dean & Ruppanner 2022] for a gender specific occurrence of stress, for my research it is instead understood as a general measurement for the stress and concentration of the user during a task. As such, the mental load can vary between a low state of either disinterest or by engaging in low effort, repetitious tasks or a high state for stressful necessary concentration. It has to be understood in my work as a form of expression of cognitive load, for example in the form of accumulating (mental) work load during increasingly complex work orders [Kindsvater et al. 2017].

Ideally the state of mental load for a user would be a continuum of values from a low to a high state and every situation in between during the naturally increasing stress of the user. Because of the highly individual perception of this state, the generation of such finely divided examples is practically impossible. It would also require the user, or an unambiguous expert, to annotate this state unbiasedly and repeatedly in the same fashion. As such, in my experiments I distinguished between low and high load, both induced by outside influence and as such non-acted, but elicited [Lotz et al. 2016]. Additionally, as the user was not aware of the induction, unwitting acting could also be excluded in this specific case.

The usage of mental load as an indicator during an interaction can be widespread. A system may assume problems and a need for further assistance in cases where the user shows a heightened mental or cognitive load, as this implies several tasks accumulating on side of the user [Melo et al. 2020]. When the system already is in an interaction, or even in a human-machine supported task such as in a smart factory, cognitive load precisely indicates to the system when the user is overwhelmed either indicating a possible slowing down of the processes or a better explanation from the system concerning the necessary support. Together with an emotion recognition from the last section, the system can also give a more “objective” measure for the current condition of the user.

Thus mental load, as used in my research, is an important aspect for continuous HMI, as it measures situations when the technical side may increase its assistive function. Differently from the former aspect of emotion it is also more relevant in a “productive” setting, in the sense where a human and a machine work together to achieve a specific result [Yang et al. 2022].

In this thesis this was already presented, with the experiment fully explained in Section 5.2.1, where the examined problems were similar to the ones described here, specifically in the context of time based decision making and increasingly hostile assistive behaviour by the technical system. The question stemming from this research is, if it is realistically implementable and practically valuable in the assistance system to provide an input on the state of the user from this information? As it can be seen in Table 7.2 high mental load recognition results

**Table 7.2:** The table list my classification results (UAR and Unweighted Average Precision (UAP)) for mental load states and is abridged from Table 5.1. The data is taken from [Weißkirchen et al. 2018].

| | Full Feature Set | | Reduced Feature Set | | |
|---|---|---|---|---|---|
| | SVM | RF | SVM | RF | |
| UAR | 69.5 | 75.5 | 68.4 | 69.8 | In |
| UAP | 69.2 | 75.6 | 60.2 | 69.9 | Group |

were achievable, indicating not only that the induction of mental load states provided robust feature differences over several speakers, but also that the used classifiers proved to be adequate for this task. Additional effects and affects used in the experiment, such as interest (for the task) or attention (without high stress) were also tested in one specific experiment, but only provided results around 55% each, i.e. slightly above chance level. Importantly, neither the dataset nor the classifier itself was specifically adapted for this experiment, as such the low result is explainable.

Given the encouraging results for the measurement of mental load taken only from vocal interactions between humans and a technical system, this also could easily become one of the standard methods for indirect user state detection. As mentioned, situations during a HMI can easily become stressful, either due to external factors or the type of assistive function the system provides. With a lack of proper measurement, these in turn can easily lead to user dissatisfaction or even potential termination of the cooperation altogether. By enabling a system with the necessary insight, relevant countermeasures can instead be taken to suppress such occurrences.

### 7.2.3 Physical Load as User State

The last inner human state to be examined in my research is physical load. It is similar to mental load in the sense that it describes the currently felt personal effort of the user during a situation. Instead of measuring the mental perception of the user, this value describes the general physical requirement as dictated for example by the personal stamina or strength of the user [Weißkirchen & Böck 2018]. With this aspect the requirements for a technical interpretation are even higher, as it is not directly expressed through an affective or even directly subconscious change of the user, but the bodily reaction and its influence on the spoken utterances themselves.

When using the state of physical load, it practically receives an indirect measurement for the general physical state of the user. This includes, similar to the mental aspect, short term problems such as sudden strains and overexertions during daily activities, requiring direct engagement by the system to identify and solve the problem. It also includes long term interactions on a task, for example during an assisted workflow or during a supervised exercise, indicating the approaching limit of the user capabilities and allows for the reduction of the user load beforehand by the system.

The physical load is complementary to the mental load in that is presents the remaining capacity of a user to increase their activity and that a potential overload by the user could negatively impact further HMI situations. Beside the importance during productive assistance situations it can also be of high importance for users who are injured or with reduced physical capabilities [Kumar & Lee 2022]. As it is an individual measurement of the specific user instead of a generalisation, it allows the system to approach these users without leading to potential further complications resulting from physical overexertion.

In my research the specific physical load was taken from elderly people on a physical training course, which also shows one of the possible applications of such a state detection. As physical capabilities strongly differ between people [Cooper et al. 2011], it is of high importance that a system would be able to recognise

**Table 7.3:** The table shows our classification results for physical load recognition. It is an abridged version of Table 5.2. Result taken from [Egorow et al. 2019].

| Data | Females | Males | Overall |
|---|---|---|---|
| - High | 83.5 | 71.7 | 79.72 |
| - Low | 83.3 | 67.4 | 78.73 |
| - UAR | 83.4 | 69.7 | 79.22 |

such capabilities independently from the user, who may be inclined to lie about their own physical disabilities or are not fully aware of their own limits. The specific experiments where explained in greater detail in Section 5.2.2 for the approach using an autoencoder and in Section 6.2.2 where a CNN was used. Without repeating the specific experiment in this section, the achievable result were very good, as seen in Table 7.3, and allowed for a physical load detection using only audio cues. This is especially interesting, as it shows the potential to measure physical load information indirectly, in this case through speech, which otherwise would require body sensors, with a varying degree of intrusion, or declaration from the user. This opens a variety of new possibilities, with the caveat that the best results for the used system was done on speech data, which is not necessarily always given during physical exertion, such as sports. Visual sensors, or the aforementioned body-sensor, could in these cases provide a more continuous mode of supervision.

## 7.3 Other User States

The former section dealt with an area of user states I described as inner states, this was done to exemplify their originally complex interpretability as there is no direct method to look into the "inner" workings of a user, the area of such states is of course even wider than the examples given in this thesis and examined in my research [Siegert et al. 2022b]. Different from these aspects, I would also define a comparable area of "outer" user states. In this area falls very roughly all information which can be recognised by an external observer. One such aspect which was often mentioned in this chapter is the idea of indirect measurement or recognition. Similar to methods in agricultural research, where indirect sensors are employed to reduce stress on organics [Zheng et al. 2021], an integrated human-machine environment could potentially prefer indirect measurements to reduce intrusion on the user. In my research, I experimented several times with such approaches, replacing an otherwise necessary visual sensor. This can be done to reduce overtly privacy concerns, such as the feeling of being watched, but also to ease the complexity to implement a semi-mobile assistant system, where either the user or the assistance system can be placed at arbitrary places in the environment.

One important example of such an approach is the measurement of addressee detection without the use of explicit information. Usually this would be done through a wake-work, or by gaze tracking in conjunction with an interaction agent. While both methods work on a certain level, both require either an unnatural action, such as a wake-word for every interaction, or clear gaze recognition, as well as a user who tends to look towards their interaction partner. By using voice based indications, which are subconsciously generated during any natural interaction, such indicators can either be supported or even fully replaced.

The reason to survey this specific aspect in my research is given by the high unnaturalness and error rate for current system addressee situations [Siegert et al. 2022b]. Wake-words both impede the flow of conversation and are prone to false activations when uttered in non-optimal circumstances [Siegert et al. 2021]. At the same successful addressee detection is also a necessity for all user-controlled interactions during a HMI and as such a major part of what a user can perceive from the system, especially in regards of functionality. The research done in this area also led to a patent employing the results taken from [Siegert et al. 2021] which used this Continuous Learning Framework (CLF) as a means to solve addressee detection failures in HMI systems. As can be seen in Table 7.4

**Table 7.4:** Results from our voice based addressee detection, abridged from Table 5.3 with information taken from [Siegert et al. 2021].

| | UAR [%] | abs. Δ | UAP [%] | abs. Δ |
|---|---|---|---|---|
| Meta classifier | 62.70 | 9.13 | 63.20 | 9.85 |
| CLF | 85.77 | 32.20 | 85.67 | 32.32 |

significant results were achieved in the detection of addressees when only using paralinguistic features. As this experiment builds the base for a proactive system design, it will be explained in greater detail in the next Chapter 8.
Using this information about the user, as mentioned, provides the basis for a more human like interaction from the side of a technical system. Otherwise such a system would have problems to approach the aspect of empathy or understanding

a human interaction partner could provide [Ephratt 2011]. It is also an important aspect of the generation of context sensitive rules beyond the scope of current user profiles which often lack deeper understanding of the reasoning behind the decisions the user takes [Champney & Stanney 2007].

## 7.4 Summary of Chapter

In this chapter I presented the utilisation of the formerly presented machine learning methods in the recognition of only indirectly measurable user states. Emotions, mental/physical load, affect, or similar states provide an important part for more natural interactions between technical systems and human partners. It was shown how each of these states could be recognised from purely speech based information and what underlying influence on the user they had.

The specific effects of such information, either used in a rule-based framework where each user state is mapped to a technical reaction, or as an input for a more natural human-like behaviour control will be examined in more detail in Chapter 9. There a (human) peer-like architecture will be presented and it will be examined how such information is integral to its function.

The next step of the framework follows in Chapter 8 where first I will explain how an assistant system can change from a purely reactive mode of operation into a proactive system, and which benefits such a mode has. With this I also go into the next layer of the assistant system framework, as described in Section 1.2. As this aspect requires the information from all the former chapters, information extraction, interpretation and indirect conclusion, all based on the machine learning framework, this will also be the finalisation of that step.

Chapter 8

# Change to Proactive Engagement

## Contents



In the last chapters we examined the general data flow of the incoming information through a data pipeline of a typical assistance system and how this information can be processed. Given that such a system is generally constrained by only reacting to the decisions of the user, it can only work with provided information. In addition to this method, the projected approach has to provide reactive and a proactive information gathering capabilities in such a situation when it is necessary [Siegert et al. 2021]. A logical continuation of this distinction provides also a connection to the following chapter concerning true peer-like technical architectures. These are ideally human-like, equal participants in all their interactions and capable of performing their own information gathering tasks.

Contrary to this theoretical high-level process of proactivity, there is also a technical intermediate step of reactivity/proactivity possible, which will be thematised here. This level of activity concerns itself primarily with the ability to collect and generate information in case of imminent uncertainty. Specifically, when a system is unable to apply a previously trained or programmed solution, either through the lack of available information or novel problems, it may discuss this directly with its user to solve the uncertainty. As the currently typical reactive approach depends mainly on the available training examples, learned before an interaction, a more proactive engaging system has to learn organically throughout an interaction additional information, as well as through an interactive human-machine rapport when required. For this a suitable method was developed by colleagues and me, which implemented this underlying idea and its self-validation in a technical way which was published in [Siegert et al. 2021] and furthermore patented as a novel method.

The basic idea for this chapter is taken from publications like [Siegert et al. 2021; Siegert et al. 2022b] as well as further initial experiments into this topic. This chapter begins with a more general analysis of the changes happening during a shift from a reactive towards proactive approach in Section 8.1. The main researched applied aspect is in Section 8.2, which explains this approach on a specific problem which is part of the Human-Machine Interaction (HMI) space, namely the addressee detection, which is in turn divided into the solving general problem in the usual reactive manner in Section 8.2.1, as well as the transfer into the proactive method in Section 8.2.2, with its measurable advantages. In Section 8.3 the method gets expanded, to further include human-machine environments, or smart environments, where human and technical agents act alongside each other in close proximity. Section 8.4 is the summary of this chapter and presents also an implementation in the general framework.

## 8.1 Different Levels of Engagement

The basic framework for a general HMI based assistant system was described in the former chapters based on a few functional groups. These can generally be understood as the interface, the interpreter and the control unit. This separation, which is also mirrored in the proposed assistant system, as seen in the illustration in Section 1.2, is functionally the same, independent from the system complexity. An important aspect during the process of designing a more advanced form of assistance system is the continuous function during situations with missing information, specifically how the system recognises and increases the awareness about a certain topic or task when it lacks the necessary background data.

Most assistant systems currently work on a voice based interface model [Grünenfelder et al. 2021]. Every action and decision is initiated directly by the user, who commands the next action by interacting with the system when assistance is wished. After receiving the initiation, often done by a specific phrase or keyword, the system parses the command and then activates the relevant technical interfaces, using databases, smart applications or similar extensions of the assistant system itself [Ricquebourg et al. 2007]. The relevant part here is the requirement of full responsibility by the user for every action taken, as each action is dependent on the initialisation done beforehand.

Contrary to this approach, an active or proactive system, as envisioned in our research in [Siegert et al. 2021] or generally as a more active companion or partner as in [Biundo & Wendemuth 2017], does not always need to wait for a user initiation, but may instead request user input themselves or even autonomously decide for the next action to happen based on other stimuli, with the intention to follow the original user intention indirectly. Such a system would not only be much more natural and efficient in its performance, but it could also be necessary for the solution of certain imminent tasks, where a purely user-activated system may prove disadvantageous. Such examples can be the care of elderly people with inhibited communication skills or coordinating operations in dangerous work environments without continuous interaction. To facilitate a system to be proactive, it must be capable to interact on its own accord and to make its own decision, using the interface and decision parts of the pipeline more independently from the user than usual.

The first aspect to examine for an active engagement in most HMI capable systems includes all potential ways in which the system and the user may declare their intention to interact with each other. The usual method in current voice control systems, classify their interaction based on speech as either device or human directed [Siegert & Krüger 2021]. Similar less naturalistic interface options use textual input or haptic surfaces, such as keyboards or touchscreens. Through current text-to-speech and speech-to-text software these forms are often directly convertible [Mache et al. 2015]. Chapter 7 described how this user based information became usable for a system, but did not describe the process to detect the addressee correctly.

The second aspect is normally lacking in current frameworks, the system based intention to engage in an interaction, which is dependent on the decision making process of the technical system. While in a current usual application a system does not need to make any decision, or at most only in the way on how to solve a command, a proactive system needs to be more involved. Based on a situational indicator and process oversight, the system needs to recognise a situation when the current task is either no longer possible to solve, or the situation as such is not known, or alternatively changed considerably from the initial situation.

Following that recognition, the system has to actively engage the user to solve this lack of information or correct the approach for the designated task.

An engaging, in addition to an active, system now can employ both aspects to recognise a problem and then interact with the user or the environment to solve the current lack of information. Additionally, when the system recognises the situation in advance, it may even attempt to solve or prevent a problem before it arises and reaches the user [Weißkirchen et al. 2020a]. This is similar for user independent tasks, which will be explained further in Chapter 9, when the peer-like level will be introduced.

The prevalent topic in this chapter is the initial solution for lacking information, and how the system can engage to generate new information. Connected to this approach is the idea of a Continuous Learning Framework (CLF), not only for a system to be able to recognise new problems, but most importantly as part of the solution to learn from the generated information through the interaction. Otherwise, without self-directed learning the system would always engage the same problem with the same solution, repeating its errors from the beginning. This ideal approach will be presented initially in this chapter as a middle step, which will then be further expanded with a full decision making aspect in the next chapter, allowing for a truly proactive system.

## 8.2 Addressee Detection in different Engagement Levels

The exemplary task examined in this chapter for a proactive engagement system is from the area of addressee detection [Siegert et al. 2021]. In the context of a voice controlled assistant system this means the ability of a technical system to distinguish utterances aimed at the system from any other sound recorded on the external sensors, such as background chatter or exclamations. Especially in a natural, or even an open microphone, environment this is an extremely relevant part of an efficiently functioning system [Richey et al. 2018]. In this section the development from the actually used method, which is comparably primitive and stilted when compared to a natural interaction, towards a more proactive and natural method, which is based on active system engagement, will be discussed. Importantly, this also provides a blueprint for other complex information dependent tasks, which otherwise require a great amount of pre-existing knowledge on the side of the system when solved with alternative approaches [Weißkirchen et al. 2020a].

### 8.2.1 Development of Addressee Detection

The general task of an addressee detection system encompasses the recognition of all possible aims, or addressees, which are possible for a speaker to have. Specifically, in case of a voice controlled assistance system, the primary motivation is the separation of device-directed speech, which is information which must be parsed, from all other recordings, often human-directed speech in a multi-user environment or similar unaimed exclamations, which can be ignored by the system or even should be ignored for privacy reasons [Guhr et al. 2020]. In real world applications this none-device directed speech can also contain background noise or self-talk and exclamations [Siegert et al. 2021]. The correct functioning of this classification is an important part of the perceived reliability of an assistant system. It also contributes to the security of a system, as false activations are often a source for significant problems for the user, especially if the used system also contains commercial applications such a buying objects trough voice commands [Liptak 2017].

As a consequence, the aspect of addressee detection is of high importance for the proposed assistance system as well as part of a proactive engagement method. To solve this issue, the general method developed into an array of different solutions. The typical avenue with the ordinarily best results is the implementation of multi-modal feature solutions, especially gaze detection and visual orientation as indicator for the current addressee of an interaction in conjunction with voice and keyword spotting [Siegert et al. 2021]. As mentioned beforehand, visual recognition is not always possible or preferable, either because of environmental or privacy reasons [Guhr et al. 2020].

The alternative which is both more natural, and less invasive, is the idea of using the speech signal itself as an indicator for the addressee, specifically to distinguish device- and human-directed speech, as also proposed in our research in [Siegert et al. 2021]. Contrary to the common method of using wake- or activation-words as in [Siegert et al. 2021], which have to be parsed to be understood, the inflection and paralingual features themselves should and can be used as base for the classification [Siegert et al. 2021]. While such a system would continuously scan the spoken utterances in its surrounding, similar to a wake-word parsing approach, it would only process the sentences if the speech signal prosody can be identified as device-directed. Such a method can be used either independently or in conjunction with other classifiers. Depending if the privacy or security requirements of the user are of a higher importance in the current situation it can also perform locally with high accuracy, as shown in the next section. The development of this approach is based on similar works as seen in Table 8.1,

**Table 8.1:** Comparison review of device-directed(DD)/human-directed(HD) recognition performance of selected studies, compiled regarding the utilize dataset and method. Specifically examining experiments using purely voice based methods. Taken from joint work in [Siegert et al. 2021].

| Reference | Measurement | Value[%] |
|---|---|---|
| "Conversational Browser" | | |
| [Shriberg et al. 2012] | equal error rate (EER) | 12.63 |
| [Shriberg et al. 2013] | EER | 12.50 |
| "Trivia-Question Game" | | |
| [Tsai et al. 2015b] | EER | 16.39 |
| [Tsai et al. 2015a] | EER | 13.90 |
| [Vinyals et al. 2012] | EER | 10.80 |
| Smart Web Video Corpus (SVC) | | |
| [Batliner et al. 2008b] | Accuracy (ACC) | 74.20 |
| [Pugachev et al. 2018] | Unweighted Average Recall (UAR) | 78.00 |
| [Akhtiamov & Palkov 2018] | UAR | 80.00 |
| [Akhtiamov et al. 2017] | UAR | 82.20 |
| [Akhtiamov et al. 2017] | UAR | 82.80 |
| Amazon in-house dataset, using [Mallidi et al. 2018] as baseline | | |
| [Mallidi et al. 2018] | EER | 10.9 |
| [Mallidi et al. 2018] | EER | 5.2 |
| [Mallidi et al. 2018] | EER | -35.36% (rel.) |
| [Tong et al. 2021] | EER | -41.1% (rel.) |
| Voice Assistant Conversation Corpus (VACC) | | |
| [Siegert & Krüger 2021] | UAR | 81.97 |
| [Akhtiamov et al. 2019] | UAR | 90.10 |
| Restaurant Booking Corpus (RBC) | | |
| [Akhtiamov et al. 2020] | UAR | 62.80 |
| [Baumann & Siegert 2020] | F1-score | 65.50 |
| [Siegert et al. 2021] | UAR | 85.77 |

and developed comparably slower than the multimodal approach but over a long period of time with different feature sets as base. Common observations between these experiments noted the high individuality in the relevant expression feature changes [Siegert et al. 2021]. Specifically, the difference between device-directed and human-directed speech was noted, but was described as quite difficult to distinguish even for human listeners and annotators. This separates this problem from the one in Section 5.2.1, where the difference was based on the biological changes in the generation of the individual speaking style of different users. Here for addressee detection the speaker behaviour or idiosyncrasies are instead the primary differentiable aspect. Common descriptions by human annotators were often contrasting, such as: More acted or more colloquial, faster or slower [Siegert et al. 2021].

To improve the results of such a classifier a more complex approach is necessary, such as in our research in [Siegert et al. 2021] where several different classifiers were combined into a metaclassifier, using the different strengths of the individual systems to support each other for a better final result. While typical methods, and even human listeners without a textual understanding (due to a language barrier), were unable to generate good classification results, our group achieved better results with a meta-classifier as seen in Figure 8.1. Interesting is the result, that human listeners by themselves are only capable to achieve chance level, while technical systems could surpass this boundary. Combining several approaches into one system provided the system with the necessary capabilities to distinguish speaking style and features more accurately. Even then, the systems were generally limited to certain levels of results, which were below what a wake-word or even multimodal approach could accomplish. To solve these restriction required a different approach to the problem, specifically a system which actively engages its user during uncertainty and learns during activation from its own errors instead of following the typical pre-trained method for classifiers [Siegert et al. 2021]. Practically this resulted in an improvement from 60.54% UAR or 53.57% UAR from human annotators, depending if they understood the language of the utterances or not respectively, to an UAR of 85.77%, even above the results possible by more complex metaclassifiers of 62.70% [Siegert et al. 2021].

### 8.2.2 Proactive Addressee Detection

The highly individualised expression of each possible user, together with the potential change of speaking behaviour during the ongoing use of an assistant system, lead to a system which may continuously lack the most current information. With a purely reactive system this is only solvable through simplifications, such as wake-word or push-to-talk activations, or intrusive measurement methods such as continuous camera surveillance [Siegert et al. 2021]. A proactive system, as proposed in [Weißkirchen et al. 2020a], instead attempts to solve the lack of

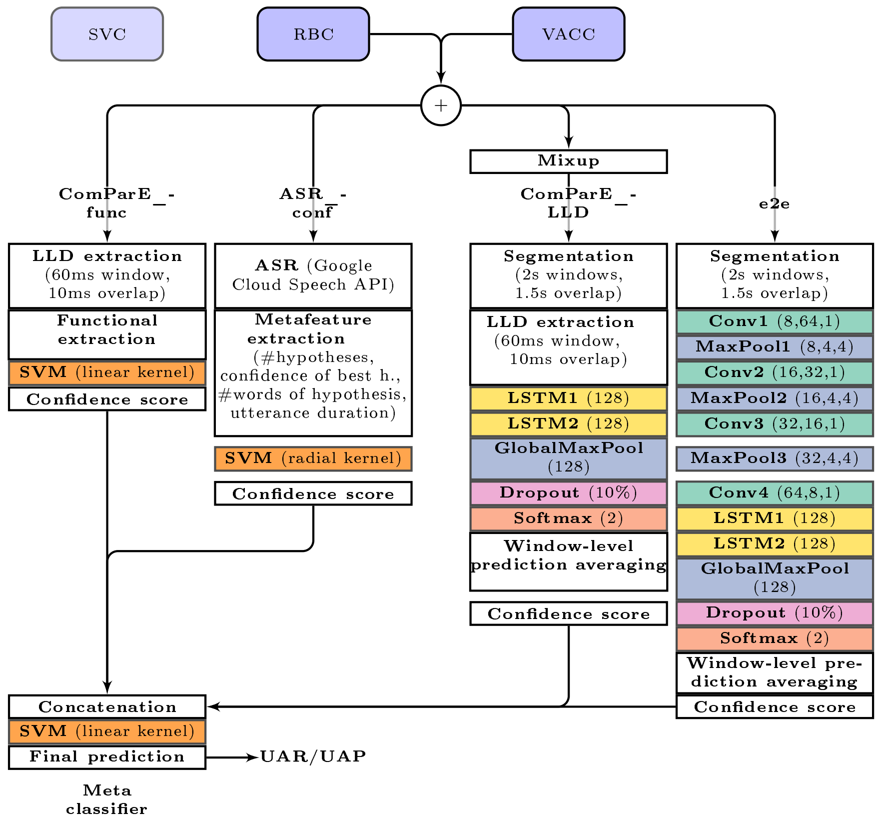



**Figure 8.1:** Architecture of the comparable meta classifier. As input, RBC and VACC feature sets were used, SVC turned out to be not as powerful. Models from left to right: ComParE_func, ASR_conf, ComParE_LLD, e2e. Notation of convolutional layers: layer name(n units, filter size, stride). Other network layers: layer name(n units). Taken from [Siegert et al. 2021].

knowledge by informing the user and generating new and relevant information as needed. This connects the system also with a CLF [Siegert et al. 2021].

The CLF describes a machine learning method, which is both initially trained and can be further adapted, especially when new or changed data is available. During any interaction new examples are created potentially, under the assumption that the system may be able to check its own result with the objective truth of the user intention. At the same time it requires for its efficient implementation that the training works iteratively. Both requirements pose certain architectural requirements not all methods can solve successfully.

The compared meta-classifier, but also usual deep learning architectures, requires often complex training stages and a lot of training initially [Siegert et al. 2021]. A change back into the training stage, after completed training, is not easily done and as such an improved or increased training set would require a new retraining of the system itself. The commercial variants solve this by employing the new training externally and only applying the trained classifier as update on the system itself, alternatively the classifying itself is often externalised on connected clusters, requiring the system to be always connected to the internet to process the otherwise local information [Chung et al. 2017]. The relevant change in the system itself to generate objective and correct training data additionally would require a local change of the system.

The solution proposed was as shown in Figure 8.2. Utterances by themselves can be recognised through volume and pauses. Instead of parsing for the existence of a wake-word, the system uses its pre-trained classifier which is based on a generalised training set of different speakers. A positive addressee result will be sent to the relevant parsing system while a negative result will be ignored. Relevant to the usual model is the situation under uncertainty or after wrong classification. In case of uncertainty, which means a result which is neither close to the category “device-directed” nor to the “non-device-directed” the system engages by simply asking if an interaction was intended. Specifically during the first interaction between user and system, where an adaption was not done, this will most likely be the main situation. Compared with current typical systems, this is already a more robust method, as the usual solution is to simply ignore unclear or ambivalent results [Siegert et al. 2021]. The questions by the system are not only used to correctly start an interaction with its user, but also to gather a label for its continuous training process. As the user themselves are questioned this would be as close to an expert labelling as a technical system can provide. The problems described before for the CLF in 5.3 are still given. The characteristics of the reoccurring user potentially develop into an overfitted system during the training, especially when no data augmentation and optimisation steps by the system are performed. Additionally, a self-learning system in such an configuration is biased by only receiving a very small amount of data, in contrast to a typical system employing user data from a wide variety of sources. The usefulness of this approach was shown in Table 7.4, where this approach was successfully implemented and capable of improving both UAR and Unweighted Average Precision (UAP) around 20% compared to the metaclassifier.

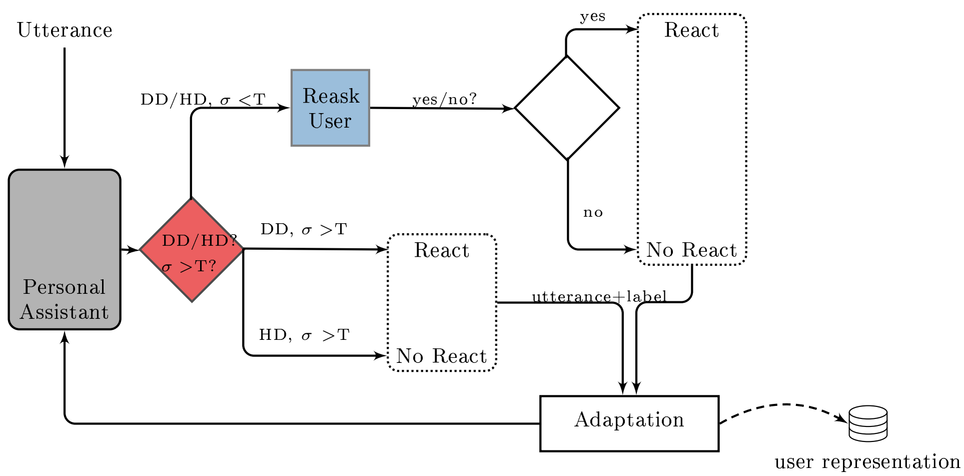



**Figure 8.2:** Scheme of our proposed Faultiness Learning Framework for improved Addressee-Detection. T defines the confidence threshold before the system engages the user. Taken from [Siegert et al. 2021].

## 8.3 Engaging Environmental Awareness

The presented solution was concentrated on the generation of further training data during a continuous interaction with a user. The potential for a system which is able to actively engage when its decision making is still uncertain is much wider, especially when in a more integrated smart environment. The interactions till now were based on the spoken utterances, as they provide the usual main interface between the human and a machine. In a smart factory, for example, with a high amount of background noise and movement, this could become less relevant [Lee 2015]. As those environments would be less concerned with privacy, visual sensors would again become more relevant.

Similar solutions are also possible in the areas of disaster relief or maintenance, especially when direct human oversight is not continuously possible [Rejeb et al. 2021]. Connecting all these different tasks, is the ability of a technical system to recognise the information level for a current situation and to independently decide if the level needs to be higher for a continuous assisting function of the system itself.

A further aspect of an engaging system is the perceived reliability and care such a system would provide [Alharbi et al. 2019]. As mentioned, a reactive system is constrained by the activation through the user, which is contrary to the idea of a continuously caring assistant which should be able to provide uninterrupted help, especially when the user is either unaware or unable to activate the necessary functions, similar to a human peer.

## 8.4 Summary of the Chapter

In this chapter we applied the formerly described information pipeline in a practical application for an assistant system, importantly in a different manner than it is usually done. While the typically employed methods can achieve good classification results, it is clear that without increasing and adding to the capabilities of the assistant system they are held back in their further development towards user adaption. With the change from the reactive and only observing approach towards the proactive and engaging method, a system is no longer constrained to its initial state. Specifically, in the area of an assistant system there is a lack of comparable developments, even though many aspects would benefit from the more active engagement behaviour, as it allows the system to solve situations under uncertainty which can easily appear due to the highly individualistic behaviour occurring with different users and the potential wide variety of situations a modern integrated HMI architecture may encounter.

The solutions presented allow for an intermediate step, where a system may still follow the typical rule-based approach of predefined reactions, but is also capable of improving its capabilities in a real-time environment. This is in contrast to current methods which are based on externalised control software, where all new abilities and applications need to be trained for a most generalised solution and most user specific adaptions based on localised user profiles which do not change the underlying classifiers and architectures.

In the next chapter this idea will be expanded further for this work, when a technical system is not only actively engaged during its interactions to a user, but equal in its status and decision making as well. Additionally, to fulfil these requirements the final control layer of the presented architecture will be explained in detail and the full data pipeline completed.

Chapter 9

# Systems as Peers

## Contents



In the penultimate chapter of this Thesis, the presented concept of a peer-like assistant system will be finalised. Following the underlying structure of the architecture, as shown in Section 1.2, the methods in this chapter also coincide with the necessary functions of the topmost layer. Specifically this includes the control and decision making capabilities of the system. As such, it is a thematic continuation of the last chapter, where the system was controlled by semi-autonomously chosen actions during a situation of lacking information. Missing in the aspect explained there was the ability to perform a truly user independent objective based approach, which will now be examined here. The idea behind this method is primarily based on the former research of me and my colleagues as found in [Weißkirchen et al. 2020a] and [Weißkirchen & Böck 2022], which is called the "peer"-level of a technical agent. This name is used to describe the relative position of an advanced technical assistant in a human-machine environment. This is primarily described in contrast to the current frame, where the technical aspect is clearly in a user subordinated position, providing often only a user interface.

The chapter is separated in different subtopics, explaining this overarching concept in greater detail. First, the general differences, peculiarities and consequences of this approach will be roughly expounded in Section 9.1. Because of the implicit impact such a system would have on its human counterpart, this is separated into a purely technical discussion in Section 9.1.1 and an observation of the higher impact and responsibilities such a system would have, specifically from an implementation standpoint, in Section 9.1.2. The architectural design for a technical implementation of such a "peer"-level system will be explained in Section 9.2.1, followed by an exemplary discussion of the transfer from the proactive/engaging system in the last chapter into this framework in Section 9.2.2. Finally, the potentially finalised assistance system will be examined in its completeness together with the summary of the chapter in Section 9.3. An important distinction of this chapter, in comparison to the former ones, is its generally conceptual nature, as the described architecture is not yet implemented into an experimental setup and its functional analysis is based on the combination of several proven subcomponents instead of a full implementation.

## 9.1 Peer Level

The name "Peer" was chosen in [Weißkirchen et al. 2020a] to distinguish a system capable of formulating and following its own set of objectives and instructions, independently from a directly supervising human user. While this may appear contrary to one of the primary tenets of a companion or assistant system, which is the idea of supporting the user during problematic tasks, it is a necessary step to remove the user based initiation from the provided support. The heightened independence allows a system to predict and solve potential problems for the user, before they become too imminent or severe which could happen if the user lacks information compared to the system. Following such a design concept, care has to be taken that such a system is not fully subsuming the decision making processes of the human side in the shared environment, as well as to find a balance between the influence of the system and the self-determination of the user.

While the practical implementations will be discussed in detail in Section 9.2, the implications of such a system will be presented in this section. For this the general impact on the user is discussed in Section 9.1.1 and the heightened requirements on the system will be examined as follows in Section 9.1.2.

### 9.1.1 Difference in Status

The idea of a "peer"-like system is a continuation of the proactive principle examined in the last chapter. The biggest difference is given by the controlling decision making process such a system must employ. The formerly described proactive engagement was controlled by a comparably simple if-then rule, such as low confidence in the generated results or lacking information of a situation as initiation. Building on this principle a "peer" would need to be able to distinguish not only these situations, but may additionally prioritise certain information or situations above others depending on its own objectives and priorities. These objectives are in turn dependent on the overarching aim of the designed system. As a result such a system would be much more human-like in its decision making process and be capable of much more natural interactions, than a simpler reactive system as in [Valli 2008].

Based on the main topic of this work, such a system still needs to fulfil the position of an assistant. Notwithstanding of further potential interpretations of a "peer", this is a role which may be qualitatively improved by a higher independence of the system itself. As mentioned before in Chapter 8, the current trend of systems develops often into voice controlled databank interface, allowing for an easy user access for a variety of information and operations [Dekate et al. 2016]. While this fulfils the requirement of assisting the user, the general idea behind an assistance system is a much wider concept, which is better exemplified as a companion and care system [Biundo & Wendemuth 2017].

By giving a technical system the ability to form independent decisions, which are neither directly controlled nor initiated by the user, the system becomes a variation of a technical agent [Strasser 2022]. This gives way for a somewhat equal human machine environment with interdependent support capabilities between the user and the system [Blackler et al. 2018]. This in turn requires a different understanding of the cooperation from both the user and the system itself. An important aspect, which will be examined in greater detail in the next Section 9.1.2, is the complications such a system may pose to a user in contrast to a purely reactive variant, and which responsibilities and safeguards have to be implemented beforehand [Meurisch et al. 2020]. The advantage is the heightened performance which is an important aspect of future designs [Ötting et al. 2020]. In contrast the current state-of-the-art is comparably low for an interactions partner, such as in Chat-bots or similar agents [Chaves & Gerosa 2021].

A specific important ability is to act even before a problem may arise or to establish positive rapport with its user. Given a more human-like design it also implicitly leads to a better understanding of the technical decision making process compared to the current black-box approaches, where the user is only presented with the results of the internal decisions [Rosenfeld & Richardson 2019]. To allow for both, an independent decision making process with its own objectives, as well as an inbuilt safety for the user from bad decisions by the system, such a design needs a robust control architecture, capable of generating, interpreting and remembering old and new rules during run-time. Additionally, to ensure that the interacting human agents may follow the processes of the system, it should also mirror the human decision processes for the most part if possible. Given the better anticipatory behaviour resulting from this, both the system and the user are more able to adapt to each other [Vinciarelli et al. 2009].

To achieve these results the proposed system should employ the so called cognitive architectures, as explained in detail in Section 2.3.2. These are methods which are specifically designed to map the human cognitive process on a technical architecture with the employment of memories and indirect rules [Kotseruba & Tsotsos 2020].

In summary, such a system is capable of operating either alone on a continuous running basis, as it finds its own tasks to accomplish based on its original starting parameter, or it can efficiently support one or several users when that is the main objective. The generation of new tasks and objectives in this case would be concentrated on this main objective of supporting the user. This is comparable with a human servant or caretaker, whose primary task is not directly stated but encompasses all occurring support situations. Such a person also needs to generate further sub-objectives to achieve this, while at the same time reacting to potentially sudden problems arising which were not predictable.

### 9.1.2 Problem of Integrated Position

An important result of a fully integrated assistant system, which is additionally capable of independent decision making, is the influence, good or bad, such a system can make on the user. While the general idea and motivation is a system with greater and better capabilities it follows that any erroneous decision the system takes influences the user to a potentially much greater degree than the typical reactive method could produce. While not necessary, it should be observable that the objective of the system is at least aligned if not actively supporting the potential human interaction partners. This can be achieved by generating a set of base rules, elaborating where and where not the full range of independent decision processes is allowed.

As a result the system should ideally be planned to generate a "self"-chosen dependency on the user. Instead of the formerly usual reactive voice command dependency, this would translate into a "desire" by the system to assure the correctness or agreeableness of its decisions with the user before the final implementation, especially when the decision is impactful on the human lifestyle [Edu et al. 2019]. Other, less impactful, decisions on the contrary would need no continuous assurance, or at least no repeating reassurance by a human operator. This is only possible by giving the system a "feel" for the relative impact.

The disadvantage of such an independent, and ideally reliable, system is the heightened dependency the user develops towards the system in question. Instead of planning each interaction beforehand, the user may be tempted to relinquish control or even high level decision making to the system. In this configuration the user would only actively engage with a problem during a phase of technical or information impasse, and may then have less necessary information to deal with this problem as they were not aware on how the system came to be. In this case a system is required to provide all necessary information before it asks for a final decision by the user, and additionally the system needs to anticipate the time needed to inform the user of the problem to prevent that the user input arrives too late.

Nonetheless, as every system is open to potential errors, human or technical, the questions of final responsibility should be cleared before using such an integrated system. For example in an industrial environment such a system would not replace the need for a human supervisor, but alleviate the workload during regular situations. In case of a home care application, similarly, the patient should still be regularly visited and examined by a capable human professional, while the technical system would primarily help with less involved day to day activities.

A final consideration has to be taken concerning the ability of any system to be too active. A system optimised for pattern recognition and proactive problem solving may unintentionally solve non-existent problems. Without an internal supervision aspect to the decision and control process, which would allow potential situations to develop to a certain degree before the correction is applied, the user could potentially become a marionette to the system. In this case a higher error rate would most likely be preferable to an overbearing system.

In effect any independent system working in close relation to a human partner needs a set of overhead rules which constrains or instructs the system on how to employ its self-steering abilities. While this detracts from the core principle of a true independent system, in lieu of the availability of morality or true understanding capabilities in current systems architectures, it is a necessary compromise.

## 9.2 Peer-like Architecture

Following the presented implications and safety measures a "peer"-like system would have, the next step is the technical solution which can potentially fulfil these requirements. This includes an architecture which can provide the general control abilities on top of the underlying information pipeline as presented in the chapter before. This includes the ability to interpret information and generate decisions based on external signals, as well as the internal implementation of user input independent objectives and priorities, while at the same time being able to constrain them for a safe user environment. In the following Section 9.2.1 the general architecture implementation will be shown, as well as a further detailed analysis in Section 9.2.2 of this new architecture in an application where situational awareness and knowledge generation has to balanced inside user-engaging frameworks.

By separating the idea of a "peer"-like assistant into the system side of decision and control and the user and environment side of information and awareness, it is easier to provide a necessary layer of security and oversight. The idea of leaving a technical system with full control over the user support, while also allowing the system to generate its own rules of solution may not only unsettle the user but can also realistically lead to grave errors as the rules may be faulty or based on correlation instead of causation. For this the system side in my approach is still regulated by easily interpretable and controllable rule-based architectures. For example, by giving the system certain boundaries for states which should not be achievable during the assistance process, certain rules can be deleted before they are applied. On the other side the abilities of current machine learning systems should be much greater than the usual generalised solution provider, both for new tasks but also for user specific adaptions, in this case the use of dynamic adaptions could become a helpful and necessary addition to provide the best possible results during an interaction. Both of these, the advantages and disadvantages, need to be evaluated against each other, which is the aim of the presented architecture.

### 9.2.1 Decision and Control

The practical solution for the establishment of a "peer"-like system as described in [Weißkirchen et al. 2020a], mirrors the general architecture from Section 1.2. The awareness the system generates about its surrounding can use the described pipeline of sensors, machine learning solutions and databanks. Included here are the directed user commands from interfaces, such as voice and haptic controls, but also indirect command information like mimic or gestures. User unattached information, such as time, temperature or similar environmental information are additionally used to generate a situational awareness, allowing the system ideally to recognise a potential overarching task or problem of the user without direct input.

In the beginning, without the required background information to solve problems independently from the user, such a system would be similar to the typical assistant system of today, with the added aspect of active engagement during task solutions. This phase is called "rule-based" in our research, because it is simply the application of standard rules, often as a result from a specific activation by the user. Importantly, during each interaction the system collects data to generate a situational profile as an experience. This goes beyond the typical user profile often generated based on repeating inquiries or interactions, as it connects user preferences with situational specifics together into a more complete representation.

Cognitive architectures allow this kind of pre-planned mapping through their inner design [Kotseruba & Tsotsos 2020]. Adaptive Control of Thought—Rational (ACT-R) for example employs a method of accessing memory units, such as specific tasks, based on the grade of matching between the remembered situation and the currently recorded one [Bothell et al. 2004]. This can be as simple as being the same command input by the user or the mentioned situational match. In the "rule-based" stage, most of these tasks are pre-programmed and connected to the control inputs. The implementation of additional peripherals also would come with their starting set of rules and regulations, mirroring again the typical design of current systems.

Differing from the typical approach is the underlying potential learning process. During any interaction or observation the system collects the mentioned data, additionally, by employing the engaging activity from Section 8.1, specific questions may be asked to ascertain user priorities and objectives. This stage, depending on the intensity of the questions posed by the system, is called the "exploration” or even “exploitation” phase [Weißkirchen et al. 2020a]. Exploitation in this case is given when the system interrupts the user in a frequency where the satisfaction and trust in the system may be reduced. Given the underlying aim of a fast and efficient user understanding this may sometimes be required and preferable instead of a continuation of the proven “rule-based” method. At the same time the system must be perceived to be trustworthy and effective to ensure a continuous use, which is needed to generate the expertise for its task, this balance act is the main objective of this part of the system [Wienrich et al. 2021].

The important aspect is the ability of the cognitive architectures to generate new rules based on experience. Particularly, the recording of a new situation combined with a system operated task can lead to a general rule which fires the task as soon as the situation arises again. These requirements before activating the task can be of different complexity, ranging from simple new command instructions to specific environmental situations independent from the former direct interaction by the user. Depending on the learning rate of a system it is either fast in the implementation of new rules or requires regular repetitions before learning a new rule.

An important aspect, especially in conjunction with the “peer”-level, which implies a certain independence from the user, is the safety and robustness of these new rules to be correct. A relatively easy pitfall such a method possesses is the codifications of simple correlations as a rule [Duangsoithong & Windeatt 2010]. Without an understanding of the underlying causation, which a technical system may not easily achieve, this is the most likely alternative result. To reduce such errors and to allow for a more secure application of this method, is should contain a series of safeguards, which were examined in greater detail in Section 9.1.2. One of the most applicable ones is the ability and requirement of the system to present a new rule to the user before memorisation.

Given a correct new design of the new rule, which is based on the behaviour and priorities of the particular user instead of the general rules available from the “rule-based” stage, which are aimed to apply to a wide variety of different users, this builds the base for the final stage. This stage is called “data-based”, and even though it repeats basically the first “rule-based” stage in its process it differs in the generation of the used rules. This difference is the adaptiveness and specific applicability of the rules on either the specific user based on real examples or on designer given assumptions based on statistical probability. During the lifetime

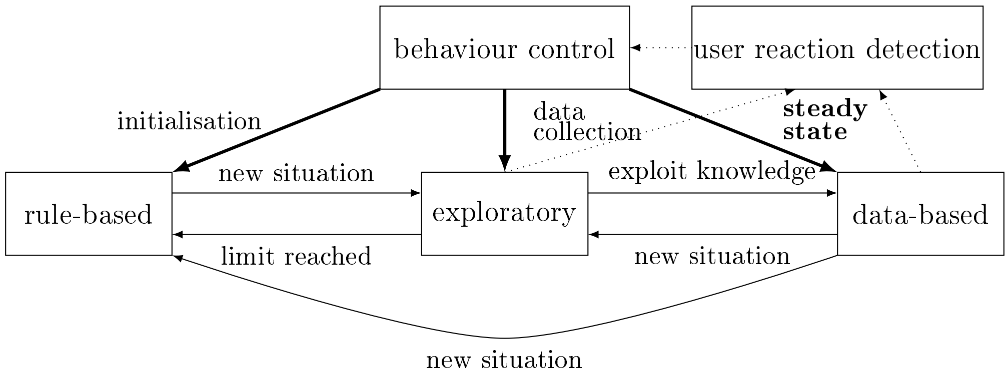



**Figure 9.1:** The architecture and control schema. The control unit observes the different stages and decides based on the change of the information and user satisfaction, which state of behaviour is preferable. The initial state is always pre-planned rules on the left. During the life cycle of the system this changes continuously towards the right. Exceptions happen when the user has to be calmed down, in which case it reverts to the simpler rule state. Figure adapted from [Weißkirchen & Böck 2022].

of the system it may continuously change between the mentioned behaviours of stable reactive behaviour, explorative information gathering and optimally user adapted rule sets. Every time a new situation arises with adequately novel information the system will change back to the exploratory stage, as it needs to learn the new parameters. In case of high complexity, and in practically every time when the user loses interest and goodwill in the borderline exploitative behaviour of the system, it may also return as a failsafe into the original rule-set. This should be done as in these situations it is simply a variant of the current architectures, combing full user control with the lowered expectations of a customarily used approach by other systems, giving a sense of stability to the user. As soon as the situation stabilises, the system may again engage in exploratory behaviour, continuously gauging the limit of the possible interaction. This can be seen in Figure 9.1 in greater detail in a technical schema.

This shows the initial decision to start the behaviour of an agent in a fully rule-based environment, where each reaction is pre-designated by the human developers of the system itself. Similar to current approaches this can be upgraded with new rules, which are generated from generalised data collections done during the lifetime of the application. These are produced externally and provide a rough approximation for the best practice over a wide variety of different users and their idiosyncrasies. As soon as the system approaches a problem or task which is not solvable by the available rules, or where the certainty of the generated solution is under a specific threshold, the behaviour shifts into the exploratory stage. This is marked by increased back and forth interaction between the system and the user, ranging from verbal or textual interviews to gathering of inducted sensor data and correlating them. Ideally during this process new rules can be designed locally, which expand or replace the generalised ones, which are instead personalised for the specific user and their singular environment. In case this explorative behaviour stresses the user, as in reaching the limit of their goodwill, the system can still fall back to the pre-designed rules from before. In the long term this method can then replace the full set of rules with individualised one, which are based on local and specific data instead of a general approximation as in the beginning. Still the ability to reduce the state of the system back into its basic form is a necessary precaution to ensure the safety of a rule set which was approved by a human observer in case of faulty training solutions. An accompanying flow diagram for the behaviour can be seen in Figure 9.2 at the end of this chapter. In case of a simple cognitive architecture, the system's reactions can instead be simulated by a basic rule set deciding the relevant changes of state. While this does not include the wide array of possibilities of a true cognitive rule system, it approximates the same reactions with less technical complexities.

When examining the system as a flow diagram one can see where the internal architecture of the system still follows a rule-based, and as such comprehensible, reaction. The adaptiveness and uncertainty concerning the decisions of the system are contained in the rules for the solution finding and not in the interaction with the user, who retains the control on which level to interact. The specific thresholds the system may employ to decide if enough data is available or if the behaviour is still agreeable can then also be decided by the users themselves. As a rough estimate for the knowledge level we proposed the amount of internal connections to other data points in [Weißkirchen & Böck 2022], while the user's agreeableness with the system can be an adaption of a typical emotion and affect recognition as presented in this thesis. The ability to discern if personalised rules approach the current problem is solved in cognitive architectures through mapping the rules on current situations.

The great improvement in contrast to the current method is the ability of the system to advance beyond the basic structure of the assistance provided. This is necessary for the greater personalisation but also allows the employment of the system in areas where new tasks may occur. The same architecture which is capable of learning from the user, can potentially also engage new environmental situations within the same framework.

### 9.2.2 Information and Awareness

In the previous section, frequently mentioned topics were the situational awareness and the information level of the system. These are technical interpretations of the human feeling of not knowing enough. As an integral part of the engagement control process it is also an integral part of the advanced idea of "peer"-like systems. Considering the change between rule-based, exploratory and rule generating stages, the confidence of the system that the available information is correct or not sufficient are a direct indicator whether the stage has to change and for the system to collect new or more conclusive data before deciding on the next step.

A purely mechanical solution would be a simply matching process, only if a situation is identical to a former state the system can assume a repetitive behaviour from the user side. Alternatively, when a situation mostly matches for several examples, a system ideally discards all information which appears to be dynamically changing between the available examples. Both these approaches employ a simple algorithm or assume an underlying simple pattern. Especially in complex human behaviour this cannot always be assumed as given, as it discards the underlying context which influences the decision. For example the changing of temperature control settings is not directly dependent on the time but more likely on the outside temperature, nonetheless examples taken from the human user would indicate that changes always occurred after a certain time as the temperature also follows a certain pattern. This is a problem in as much as the system would not directly be able to distinguish between the correlation and the causation.

To reduce the impact of correlation instead of causation, which is given when employing pattern recognition, as the system may not understand the underlying connections, several solutions are possible. The most straightforward but not often employed is by simply engaging the user, asking for direct confirmation before changing or creating a rule. This would allow for the user to reclaim the decision making process, while still allowing the system to assist in the creation of the rules. This assistance by itself would take workload from the user, who otherwise would need to program each aspect of the assistance system themselves, instead of semi-automatically as described here. Alternatively, we proposed in

[Weißkirchen & Böck 2022], the designation of a knowledge value for a specific information based on the amount of connected available information. Practically this means that a situation or topic is only assumed to be understood, when the system was able to generate a suitable frame of reference in conjunction. This can either be repeating behaviour under the same or with slightly different circumstances. In case of the last example, when only connecting the temperature control to one frame of reference, the time, the knowledge value is low and should be cleared with the user. Alternatively, when the system is able to connect time, outside temperature, location of the user and personal temperature preference together the value is high and the creation of a rule can be approached. Generally, the higher the influence on the user the rule would have, the higher the amount of knowledge should be before the creation should be attempted. The important difference to the usual approach is the continuous search for contextual information instead of a surface level pattern matching. This requires active information gathering from the side of the system, requiring a new approach for Human-Machine Interaction (HMI). Especially in cases where prior decision making processes no longer apply, instead of slightly adapting the pattern the system would now search for the underlying connecting correlations.

A possibility is to describe the knowledge value as a formulaic expression, such as:

$$K(T) = \sum_{i=1}^{n}(I_i(T) + \sum_{j=1}^{m}(w_{i,j}(T) \cdot S_{i,j}(T))) \tag{9.1}$$

where $K(T)$ is the knowledge value concerning the topic $T$, and

$I_i(T)$ are all beliefs or information directly concerning $T$, where $n$ is the amount of available information,

$S_{i,j}(T)$ are all the contextual information concerning $T$, where $m$ is the amount context information for each $i$,

$w_{i,j}(T)$ are the weighted importance which connects the context to the original topic $T$

To control the general trend of the system to collect new knowledge indiscriminately, the system needs a complementary value of user satisfaction. User satisfaction is to be understood as a fusion of different aspects indirectly explaining the general agreeableness of the user to continue with a system interaction. Practically, a user may be discouraged from further system engagement, either because of dissatisfaction or impatience with the system to cope with a new situation without existing rule. This may lead to a discontinuation of further user input, as the user decides to approach the task themselves without further assistance.

A negative user reaction towards the help received from the system is the worst case for the assistance system itself, as it may stop any interaction and in turn any further data generation, as the system is viewed as unreliable and inefficient to provide a solution. In such cases where the satisfaction drops to a low level it is of higher importance for the system to assure the user either its competence or reliability, as the base problem of lacking information most likely still prevails. The system instead should try to solve the dissatisfaction by following a proved method. In the presented architecture as seen in Figure 9.1 this is achieved by changing from “exploitative” behaviour back to the original “rule-based”.

By changing between these extremes dynamically instead of simply switching the illusion of human like behaviour is generated. The system approaches a new problem and slowly develops interest, as soon as the user is dissatisfied it tries to reassure by following simpler methods while acknowledging its shortcomings. This is not only done to better achieve a long term data generation, but also to assure a felt form of reciprocal empathy, where the user may foresee the future behaviour of the system. This gives another aspect of trust, or at least habitus between both agents. This should lead generally to better long-term cooperation than a singular approach or method, which would lock the system into a non-reactive way.

The general examination till now worked on the usual voice interaction structure, incorporating questions and answers with a higher situational awareness. As explained in Section 2.3.1, interactions can be happening without voice, either by other interfaces or simply through indirect assumptions about the interaction partners intentions. Viewing all interactions between the user, the environment and the system as sources for further information, the system may engage, on a small level, in manipulation of the environment to create further reaction from the user. For example the closing of window blinds during high outside illumination can be employed to gauge the interest of the user to make this a regular decision of the system during different illuminations levels, or at least generate a system question if this action is wished by the operator.

## 9.3 Summary of Chapter

In this chapter the discussion of the assistant system was completed with all necessary functional aspects as described in the introductory chapter. From the technical awareness through sensors in the human-machine environment till the human-like decision making processes using cognitive architectures, the system is described over all stages and in its functions as an advanced assistant system which is not only capable of reactive but instead of proactive user engagements. The final layer of the cognitive architecture as described here would allow such a system to possess a growing potential in excess of the currently regular applications. Given the full set of abilities such a system provides, it is also capable to work in an unknown situation and adapt [Jakobson et al. 2006]. This concludes the presented system as supported by my research and is followed with my conclusions and outlooks in the last chapter which is next.

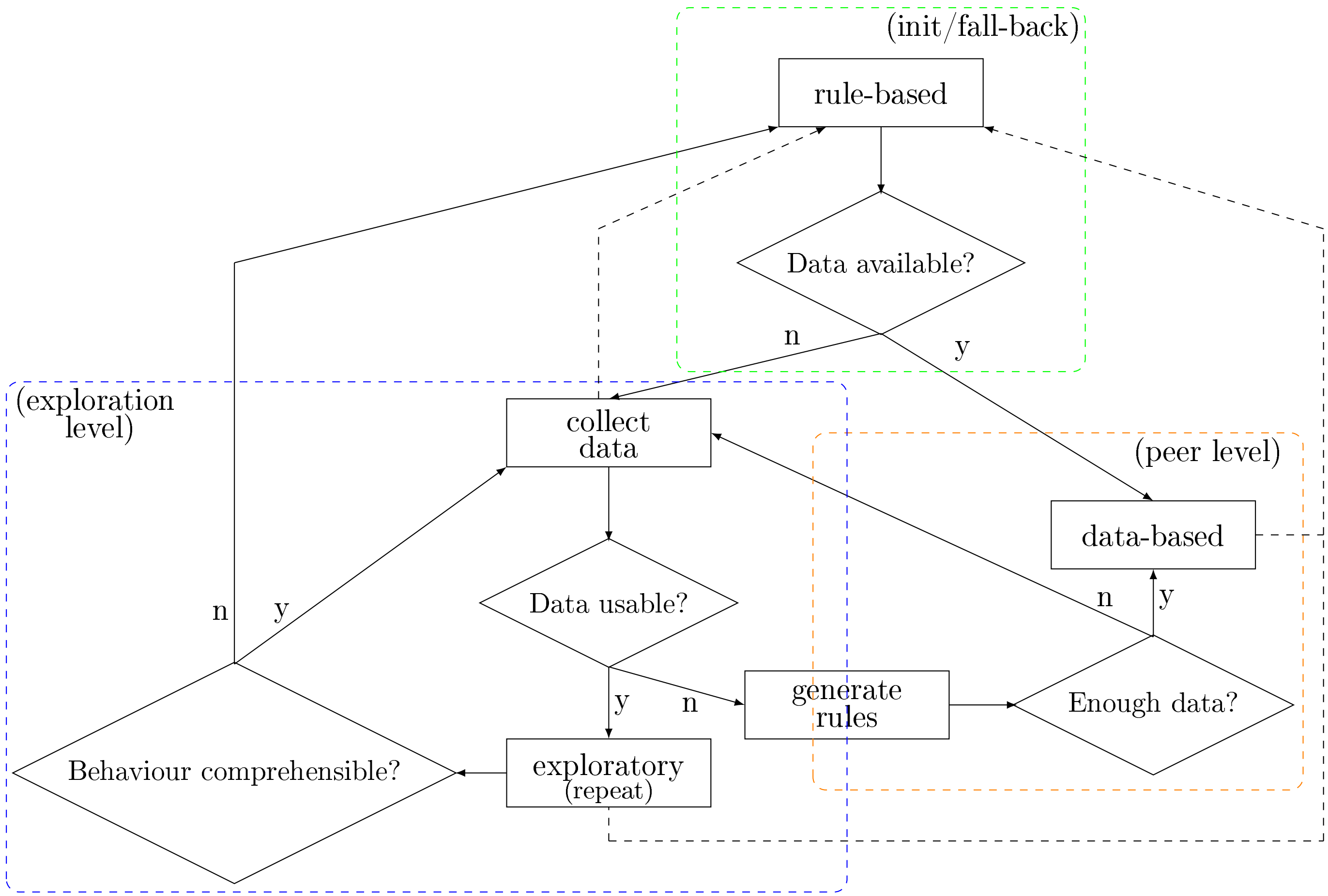



**Figure 9.2:** Representation of the inner decision flow controlling the state of behaviour. As soon as the necessary requirement is met, the system changes into the appropriate method. Alternatively the system returns to the initial rule-based state. Figure taken from [Weißkirchen & Böck 2022].

Chapter 10

# Conclusion and Outlook

## Contents



This thesis described an advancement of the typical assistance system employed nowadays. The general structure of a continuous information pipeline was described and separated into three functional groups, each presenting a different aspect of a modern system with its own included objectives and tasks. During this thesis high importance was set onto the architectural design and methodologies behind machine learning and cognitive architectures. Both were presented with typical applications and state-of-the-art implementations in Chapter 2. The other important tool used during this thesis was a selection of employed datasets, given the nature of self-learning architectures; these directly influence the capabilities of the finalised system, by either providing idealised expressions or real world occurrences. With the presented results from Chapter 4 onwards, I presented the improvements for the feature extraction and processing in Chapter 5 and the preparation for different modalities for different tasks in Chapter 6. Concerning the applied machine learning methods Chapter 7 examined the different deep learning applications employed, followed by Chapter 8 with the results for employing machine learning solutions for the detection of inner user states. Finally the high level decision making and control aspects of an assistant system were examined in Chapter 9, first as proactive information gathering device which works independently from the employed pre-training, a second as an overarching cognitive architecture control system. The results will be finally shortly collected in this chapter with a conclusion and an outlook for further research opportunities in this sector.

## 10.1 Results and Summary

During the thesis a variety of different topics were examined and potential solutions for the overarching architecture were researched. In the following the specific results will be reviewed with a deeper insight on how these results may influence the potential true "peer"-like assistance system. The underlying structure is again based in the information processing pipeline, as it defines how any interaction or situation is processed throughout the system coming from the real world, and finalising in the overarching cognitive control architecture.

### 10.1.1 Results for the System Awareness

In Chapter 4 significant features were the main objective of the research. For this the identification of these features was relevant, as well as a possible reductions of the necessary amount of data points. The importance of this research is to reduce the necessary processing power and complexity, so that even a system with a plethora of different observed environmental and user states is not overloaded and can react in a timely fashion to a change in situation. Significant features are as such defined in this context as information rich singular data points, which ideally in a collection contain little overlap between each other.
The first research done was to distinguish acoustic features which contain a lot of overlap when observed over several human speakers, but which are potentially rich in information. This is an effect due to the high dynamic and individualistic qualities of speech, especially in their expressiveness. To evaluate the effect of the significance, the capability of a classifier with the different features was taken as measurement. The result was the reduction from 933 features down to 16 which provide nearly the same classification abilities, while requiring the speakers to be roughly grouped based on their speaking behaviour, in this simple case by biological sex and age.

The second area of research, done in Chapter 5, examined the general changes when transferring acoustic data into a visual representation. This is relevant as many current state of the art classifiers are based primarily on visual input data, opening a wide variety of solutions to solving these acoustic tasks. Alternatively, it also examines the possibilities how visual data representation can contain data in several dimensions, in contrast to purely acoustic ones which are represented either in a wave format or as a simple value array. The localities of information, similar to the temporal changes, provide relevant information snippets for many tasks and as such may provide a different improvement. The research showed that spectrograms provide a dense and expressive representation for visual classifiers. It is also still relatively well interpretable for human experts, in contrast to a simple waveform depiction. It also allows for the comparison over several examples which allow the depiction of “hot-zones” or key points of information for different tasks, such as exemplarily the depiction of emotion.

In conclusion both areas of research proved positive, the high individuality of speakers may inhibit the abilities of a technical classifier, but when implementing a relatively simple pre-selection system the amount of necessary information can be reduced significantly allowing either better generalisation or the implementation of more tasks on the same system. In summary it can also be said, that changing the representation from one format to another, as from acoustic to visual, can be positive for a variety of reasons, such as information density or interpretability.

### 10.1.2 Results for the System Understanding

The next topic of research concerned itself with the ability of machine learning systems to interpret the aforementioned acoustic signals into machine representations of human user states. Because of the wide variety of different applications and architectures, this concerned most of the performed experiments to one degree or another. The experiments described in Chapter 6 were primarily about the implementation or evaluation of deep learning architectures, as well as potential alternatives possessing similar advantages without the high requirements. Alternatively it also explored the possibility of changing from a pre-trained generalised classifier to a system continuously adapting over time to a specialised selection of user characteristics.

The research into deep learning, such as Convolutional Neural Network (CNN), provided insight into a system which is capable of high classification results but also requiring a high amount of example sets and data. Alternatively my research presented the idea of simply layering classifiers after each other, using the information extracted from the significant feature research before. Using this method, a result of around 69% Unweighted Average Recall (UAR) could be achieved either with greater or smaller datasets, in the classification of the mental load in human speakers. At the same time, the layering of two classifiers behind each other improved the results compared to a singular used Support Vector Machine (SVM) system, when using the smaller feature set. This allowed the system to work satisfactory even in data size areas in which traditional deep-learning architectures are not applicable. As a disadvantage the system designer needs to decide on the different classifying functions for each layer, in contrast to a traditional deep learner which can train these functions themselves.

Alternatively this ability to optimise the feature selection for a classifier can also be implemented into a layered architecture through the use of an auto-encoder. With this implementation the presented experiment was able to generate a feature set which was reasonably robust even for different datasets using similar but not the same input types.

The alternative to the mentioned generalised and pre-trained architectures is the use of smaller but specialised systems. In my research this was done on a SVM based classifier for a problem which was too complex to be solved in a pre-training step. Instead by training the system over time with examples during an interaction with a specific speaker the system was able to achieve results in excess of 85% UAR, while similar complex system remained at chance level and more complex meta-classifier reaching 62.70% UAR.
In conclusion, the results in this chapter showed promising results in exchanging complex deep learning architectures with architectural alternatives. Especially for smaller example sets, or situation where human expertise is available this allowed for a great reduction in complexity.

When employing visual input, either because of the type of performed classification or because of the chosen representation of the available data, the use of a convolutional neural network becomes theoretically possible. This is relevant as it presents the opportunity to employ a wide range of tools and methods of a widely adopted and actively developed system, both by academia and business applications. As described before, it still requires a great data size to reduce the impact of overfitting and to assure a robust training phase, which then can employ a system which has proven to be capable of even very complex applications. In my research, which was at the time novel in its method, I implemented an emotional classifier on exactly this architecture to observe this effect.

In my work I could observe the impact of both the data size and the employed architectural design, such as depth and number of functional units. And while some tests with more complex approaches only achieved comparable results to regular baseline methods, this effect could primarily be attributed to the small data size which was employed during the training. This leads to the conclusion, that certain aspects of an integrated system should to be designed beforehand to employ the most effective use for the available data. In this case it could be advantageous for the system to develop two different approaches in parallel, one applies to the currently available data with fast optimisations and good short term results and another system which would be trained during a longer period of time and would take over the classification task after achieving stable and better results. Combined with the active engagement in my latter research such a system would produce its own training set over time with its own set of priorities based on the environment and the users.

One of the most relevant aspects to employ machine learning solutions in human assistance systems is the ability to discern information form the human interaction partner, which are not readily measurable, such as emotions, mental and physical states or unmentioned dispositions. Using the ability of self-learning instead of fully pre-designed human expert knowledge allows for better results in such high complexity tasks which are additionally highly dependent on the specific expressions of the human user. The disadvantage for all these methods in contrast is the general use of correlation and not causation in the finding of the results. As such the systems require a great amount of external and internal checking to assure that not erroneous or falsely biased training occurs, which then skews the achieved results.

In my research, shown in Chapter 7, these tasks where primarily concerned with the recognition of these user states through acoustic information, allowing for both a close approximation of human abilities through the interpretation of paralinguistic features, as well as a perceived reduction in privacy invasion in contrast to a system employing a continuous visual observation or requiring the use of wearable sensor equipment.

The examples used in my research were three different sets of user states, emotions, mental load and physical load. Emotions and the more inclusive aspect of affections describe a baseline for many interactions. As shown by my and external results the general declaration of emotional states is robust and results in good results which are comparable or even higher than comparable human abilities. As this aspect would directly influence the way a human or technical system should approach a speaker it would build also the basis for the behaviour and engagement control. The mental load, in the context of this thesis described as the general stress and concentration ability of a user when performing more mental tasks, is equally important for a system to distinguish. Especially the assistance aspect needs an independent verification of the user's mental state, given the inability to rely on an accurate self-expression of most speakers to accept when they are under mental duress. My research showed high results, especially given the ability of a system to learn the individual expressions of a specific user, comparable with a long-term user of an assistance system. The last experiment aspect is the physical load, similar to mental load but concentrated on physical instead of mental tasks. The speciality in this case is the high indirectness of the interpretation, as the most measurable indicators for physical stress are based on inner body functions and not on speech or acoustic sources. Nonetheless, good results were also achievable for this task, allowing a potential assistance system to support or alarm a user when a certain stress level is reached.

In conclusion it can be said that acoustic information in conjunction with machine learning systems can be applied to recognise a very wide variety of otherwise not measurable inner user states. Importantly, these were all testable against human expert knowledge.

### 10.1.3 Results for the Peerlike Systems

The next research, done in Chapter 8, was done to examine the possibility to change the engagement level from a technical system to get away from a typical rule-based means. As an aspect which is directly accessible from the human user, specifically how the system acts or reacts to outside situations and interaction, it is a necessary area of improvement to achieve “peer”-like reception. As it precedes the later change in control behaviour, it only employs additional elements to the formerly explained tools.

As an important aspect of engagement, the interaction success depends strongly on the ability to discern if the user attempts to communicate with the system or with another agent not connected to the technical system. This aspect, called addressee detection, works primarily through wake words or activation controls. To improve the natural interaction a system should be capable of identifying the addressee based on the intonation alone. As such the system designed in our experiment in Chapter 5 was able to learn the specific expressions of a human speaker over time, using the results from the Continuous Learning Framework (CLF). Even for such a complex problem the finally resulting system was capable of reaching a high level of robustness. For an integrated human machine environment, where the same human users would continuously interact with the system, the perceived time duration for the training would be acceptable. Such a solution prohibits the implementation in an open environment with changing human users

present, as a system would be unable to achieve similarly high results.
Based on these experiments, and the underlying idea of engagement levels, a similar system for mobile technical agents would be implementable. As a system would discern a lack of information it would engage by collecting new information and then improve its own trained abilities further.

The final aspect, examined in Chapter 9, was the change from a typical reactive architecture to a more independently deciding "peer"-like system. The important aspect of this is the ability to provide a system not only with the ability to choose from several options, but also to generate its own solutions and to react in an understandable manner for an outside observer. This part was mainly on a theoretical level. It examined two connected but discrete problems for such a system. The first part is the general structure on how to implement such a system with the realistically available technology, and the other part was a practical example for the implementation to control the engagement and information gathering for a system to improve both aspects of the system.

The first question concerning the most qualified underlying architecture was answered in our examination of cognitive architectures, like Adaptive Control of Thought—Rational (ACT-R), which is directly oriented on human cognitive functions. The presented pipeline is directly implementable in the general architecture as one of the input interfaces. Even more importantly, the ability to design methods to solve a problem and compare with older situations to then decide based on available information is the function of the memory accessing stage in ACT-R where all tasks are saved in this manner. Overarching priorities are also part of the general structure, with the ability to create new rules and actions based on these priorities when given the rights by a human observer. The important part is that, while at no stage true artificial intelligence is achieved by the system, the reaction on outside stimuli is exactly modelled after the human way. Given a priority for assistive functions, such as smart home environments or smart factories, the system would be filled with the most obvious first situations and solutions and would then continuously adapt further to the developing situation, based on occurring new information.

The examination of a practical problem presented the idea of a system which was trying to optimise the system side need for information gathering, while at the same time providing a positive rapport with the user concerning fast and efficient reactions to every interaction. Comparing this problem with a variant of the exploration-exploitation problem, the system needs to achieve a sensible balance of the two priorities, without possessing a universal ground truth for best behaviour. Depending on the current user, as well as their specific current situation, a user may be willing to interact for longer times with the system while providing all missing information required for the system to generate a complete profile of the interaction. Alternatively, during high stress and pre-existing emotional situations, a user may prefer a simple but reactive version of an interaction which provides the necessary assistance without further rapport between the user and the system. In the examination a structure was introduced which continuously observes the user state and gives a system a variety of interaction or engagement behaviour options, which changes the short-term priorities on the fly, while retaining the overarching priorities of the original base-line system.

## 10.2 Future Works

Given the results of this work, the general outline and implementability of an advanced assistance system can be assumed to be possible. Given the wide array of possible applications for assistance systems there is still a wide variety of partial functions whose capabilities could be researched. Examining the pipeline there expand several areas of interest for further consideration.

First of all in the area of awareness, there is still the expansion of all the available modalities, especially the visual and body wearables. While most systems would ideally reduce the necessary modalities to reduce the invasion of privacy, other studies have shown the positive effects of employing multimodal input information.

Concerning the use of machine learning methods, there is still the ability to discern situational instead of inner user states. This requires the use of use cases as they appear in typical assistance applications, as well as their use in external sensor applications. Further experiments, requiring much greater example sets, could be done to find the specific change from one system framework to another, specifically the deep learning one. Additionally, more usage of the continually learning framework should be done as an alternative for the most complex of classifying tasks, as to their ability to adapt to special situations without losing their general abilities.

Most importantly of all, the overarching cognitive architecture needs to be applied in a real life simulator, while all the different aspects work, based on internal and external research, it needs to be condensed into a singular system employing all the different aspects of a fully functioning assistance system. Such an experiment would need the ability to implement such a system for a long-term for it to learn during its lifetime all the abilities which otherwise would need to be fed in manually per design.

# References

## References related to the Author

### Contributions to international peer-reviewed journals

Siegert, I., Weißkirchen, N., Krüger, J., Akhtiamov, O. & Wendemuth, A. (2021). 'Admitting the addressee detection faultiness of voice assistants to improve the activation performance using a continuous learning framework'. *Cognitive Systems Research* 70, pp. 65–79.

Siegert, I., Weißkirchen, N. & Wendemuth, A. (2022b). 'Acoustic-Based Automatic Addressee Detection for Technical Systems: A Review'. *Frontiers in Computer Science* 4, s.p.

Weißkirchen, N. & Böck, R. (2022). 'Behaviour of true artificial peers'. *Multimodal Technologies and Interaction* 6.8, p. 64.

### Contributions to national peer-reviewed journals

Weißkirchen, N. & Böck, R. (2018). 'Toward a self-adapting resource-restricted voice-based Classification of Naturalistic Interaction Stages'. *Kognitive Systeme* 2018.1, s.p.

### Contributions to international peer-reviewed conference proceedings

Egorow, O., Mrech, T., Weißkirchen, N. & Wendemuth, A. (2019). 'Employing Bottleneck and Convolutional Features for Speech-Based Physical Load Detection on Limited Data Amounts'. In: *Proceedings of the Interspeech 2019*. Graz, Austria, pp. 1666–1670.

Weißkirchen, N., Bock, R. & Wendemuth, A. (2017). 'Recognition of emotional speech with convolutional neural networks by means of spectral estimates'. In: *Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW 2017)*. San Antonio, TX, USA, pp. 50–55.

Weißkirchen, N., Böck, R. & Wendemuth, A. (2020a). 'Towards True Artificial Peers'. In: *IEEE International Conference on Human-Machine Systems (ICHMS 2020)*. Rome, Italy, s.p.

Weißkirchen, N., Böck, R., Wendemuth, A. & Nürnberger, A. (2018). 'Significance of Feature Differences in the Distinction of Mental-Load'. In: *IEEE International Conference on Systems, Man, and Cybernetics (SMC 2018)*. Miyazaki, Japan, pp. 2608–2613.

Weißkirchen, N., Reddy, M. V., Wendemuth, A. & Siegert, I. (2020b). 'Utilizing Computer Vision Algorithms to Detect and Describe Local Features in Images for Emotion Recognition from Speech'. In: *IEEE International Conference on Human-Machine Systems (ICHMS 2020)*. Rome, Italy, s.p.

# References

Adadi, A. (2021). 'A survey on data-efficient algorithms in big data era'. *Journal of Big Data* 8.1, p. 24.

Agarap, A. F. (2018). 'Deep Learning using Rectified Linear Units (ReLU)'. *arXiv e-prints* abs/1803.08375, s.p.

Akhtiamov, O. & Palkov, V. (2018). 'Gaze, Prosody and Semantics: Relevance of Various Multimodal Signals to Addressee Detection in Human-Human-Computer Conversations'. In: *20th International Conference on Speech and Computer (SPECOM 2018).* Leipzig, Germany, pp. 1–10.

Akhtiamov, O., Siegert, I., Karpov, A. & Minker, W. (2019). 'Cross-Corpus Data Augmentation for Acoustic Addressee Detection'. In: *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue.* Stockholm, Sweden, pp. 274–283.

— (2020). 'Using Complexity-Identical Human- and Machine-Directed Utterances to Investigate Addressee Detection for Spoken Dialogue Systems'. *Sensors* 20.9, p. 2470.

Akhtiamov, O., Ubskii, D., Feldina, E., Pugachev, A., Karpov, A. & Minker, W. (2017). 'Are You Addressing Me? Multimodal Addressee Detection in Human-Human-Computer Conversations'. In: *19th International Conference on Speech and Computer (SPECOM 2017).* Hatfield, UK, pp. 152–161.

Albornoz, E. M., Milone, D. H. & Rufiner, H. L. (2010). 'Multiple feature extraction and hierarchical classifiers for emotions recognition'. In: *Development of Multimodal Interfaces: Active Listening and Synchrony.* Vol. 5967, pp. 242–254.

Alharbi, M. A., Alharbi, N. T., Alharbi, H. M. & Ibrahim, D. M. (2019). 'Patient Assistance System: A Proposed Structure'. In: *10th International Conference on Information and Communication Systems (ICICS 2019).* Irbid, Jordan, pp. 230–233.

Alty, J. & Guida, G. (1985). 'The Use of Rule-based System Technology for the Design of Man-Machine Systems'. In: *IFAC Proceedings Volumes.* Vol. 18. 10. Varese, Italy, pp. 21–41.

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M. & Farhan, L. (2021). 'Review of deep learning: concepts, CNN architectures, challenges, applications, future directions'. *Journal of Big Data* 8.1, p. 53.

Amara Korba, K. & Arbaoui, F. (2018). 'SVM Multi-Classification of Induction Machine's bearings defects using Vibratory Analysis based on Empirical Mode Decomposition'. *International Journal of Applied Engineering Research* 13.9, pp. 6579–6586.

Anderson, J. (2007). *How Can the Human Mind Occur in the Physical Universe?* Oxford series on cognitive models and architectures. New York, NY, USA: Oxford University Press.

Asadullah, M. & Raza, A. (2016). 'An overview of home automation systems'. In: *2nd International Conference on Robotics and Artificial Intelligence (ICRAI 2016).* Rawalpindi, Pakistan, pp. 27–31.

Batliner, A., Steidl, S., Hacker, C. & Nöth, E. (2008a). 'Private emotions versus social interaction: a data-driven approach towards analysing emotion in speech'. *User Modeling and User-Adapted Interaction* 18 (1), pp. 175–206.

Batliner, A., Fischer, K., Huber, R., Spilker, J. & Nöth, E. (2003). 'How to find trouble in communication'. *Speech Communication* 40 (1), pp. 117–143.

Batliner, A., Hacker, C. & Noeth, E. (2008b). 'To talk or not to talk with a computer'. *Journal on Multimodal User Interfaces* 2 (3), pp. 171–186.

Baumann, T. & Siegert, I. (2020). 'Prosodic Addressee-Detection: Ensuring Privacy in Always-on Spoken Dialog Systems'. In: *Proceedings of the Conference on Mensch Und Computer*. MuC '20. Magdeburg, Germany, pp. 195–198.

Bellman, R. & Kalaba, R. (1959). 'A mathematical theory of adaptive control processes'. *Proceedings of the National Academy of Sciences* 45.8, pp. 1288–1290.

Biundo, S. & Wendemuth, A. (2017). *Companion Technology: A Paradigm Shift in Human-Technology Interaction.* Cham, Switzerland: Springer International.

Blackler, A., Popovic, V. & Desai, S. (2018). 'Research methods for intuitive interaction'. In: *Intuitive interaction: Research and application.* Boca Raton, FL, USA, pp. 65–88.

Blum, A. L. & Langley, P. (1997). 'Selection of relevant features and examples in machine learning'. *Artificial Intelligence* 97.1, pp. 245–271.

Böck, R., Hübner, D. & Wendemuth, A. (2010). 'Determining optimal signal features and parameters for HMM-based emotion classification'. In: *15th IEEE Mediterranean Electrotechnical Conference (Melecon 2010).* Valetta, Italy, pp. 1586–1590.

Borst, J., Bulling, A., Gonzalez, C. & Russwinkel, N. (2022). 'Anticipatory Human-Machine Interaction (Dagstuhl Seminar 22202)'. *Dagstuhl Reports* 12.5, pp. 131–169.

Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C. & Qin, Y. (2004). 'An Integrated Theory of the Mind'. *Psychological Review* 111.4, pp. 1036–1060.

Braun, A., Wichert, R., Kuijper, A. & Fellner, D. W. (2014). 'A Benchmarking Model for Sensors in Smart Environments'. In: *Ambient Intelligence.* Eindhoven, The Netherlands, pp. 242–257.

Breiman, L. (2001). 'Random Forests'. *Machine Learning* 45.1, pp. 5–32.

Brown, W., Morris, R. J., Hollien, H. & Howell, E. (1991). 'Speaking fundamental frequency characteristics as a function of age and professional singing'. *Journal of Voice* 5.4, pp. 310–315.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W. & Weiss, B. (2005). 'A database of German emotional speech'. In: *9th European Conference on Speech Communication and Technology (Interspeech 2005).* Vol. 5. Lisbon, Portugal, pp. 1517–1520.

Calonder, M., Lepetit, V., Strecha, C. & Fua, P. (2010). 'BRIEF: Binary Robust Independent Elementary Features'. In: *11th European Conference on Computer Vision (ECCV 2010).* Vol. 6314. Heraklion, Greece, pp. 778–792.

Campbell, N. (2004). ‘Listening between the lines: a study of paralinguistic information carried by tone-of-voice’. In: *First International Symposium on Tonal Aspects of Languages (TAL 2004)*. Beijing, China, pp. 13–16.

Champney, R. & Stanney, K. (2007). ‘Using Emotions in Usability’. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 51. Baltimore, MD, USA, pp. 1044–1049.

Chaves, A. P. & Gerosa, M. A. (2021). ‘How Should My Chatbot Interact? A Survey on Social Characteristics in Human–Chatbot Interaction Design’. *International Journal of Human–Computer Interaction* 37.8, pp. 729–758.

Chen, J. & Chaudhari, N. (2009). ‘Segmented-memory recurrent neural networks’. *IEEE Transactions on Neural Networks* 20.8, pp. 1267–1280.

Chen, R.-C., Dewi, C., Huang, S.-W. & Caraka, R. E. (2020). ‘Selecting critical features for data classification based on machine learning methods’. *Journal of Big Data* 7.1, p. 52.

Chignell, M., Hancock, P. A. & Takeshita, H. (1999). ‘Chapter 11 - Human—Computer Interaction: The Psychology of Augmented Human Behavior’. In: *Human Performance and Ergonomics*. Handbook of Perception and Cognition (Second Edition). San Diego, Cal, US: Academic Press, pp. 291–328.

Chung, H., Iorga, M., Voas, J. & Lee, S. (2017). ‘“Alexa, Can I Trust You?”’ *Computer* 50 (9), pp. 100–104.

Cooper, R. et al. (2011). ‘Age and Gender Differences in Physical Capability Levels from Mid-Life Onwards: The Harmonisation and Meta-Analysis of Data from Eight UK Cohort Studies’. *PLOS ONE* 6.11, pp. 1–14.

Cortes, C. & Vapnik, V. (1995). ‘Support-vector networks’. *Machine Learning* 20.3, pp. 273–297.

Davidson, R. J. (1994). ‘On emotion, mood, and related affective constructs’. In: *The nature of Emotion: Fundamental Questions*. Oxford, UK: Oxford University Press.

Dawson, H. L., Dubrule, O. & John, C. M. (2023). 'Impact of dataset size and convolutional neural network architecture on transfer learning for carbonate rock classification'. *Computers & Geosciences* 171.2, p. 105284.

Debnath, L. & Antoine, J.-P. (2003). 'Wavelet Transforms and Their Applications'. *Physics Today* 56.4, pp. 68–68.

Dekate, A., Kulkarni, C. & Killedar, R. (2016). 'Study of Voice Controlled Personal Assistant Device'. *International Journal of Computer Trends and Technology* 42.1, pp. 42–46.

Dellaert, F., Polzin, T. & Waibel, A. (1996). 'Recognizing emotion in speech'. In: *Proceedings of the Fourth International Conference on Spoken Language*. Vol. 3. Philadelphia, PA, USA, pp. 1970–1973.

Dhall, A., Goecke, R., T., G. & Sebe, N. (2016). 'Emotion recognition in the wild'. *Journal on Multimodal User Interfaces* 10 (2), pp. 95–97.

Dojchinovski, D., Ilievski, A. & Gusev, M. (2019). 'Interactive home healthcare system with integrated voice assistant'. In: *Proceedings of the 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO 2019)*. Opatija, Croatia, pp. 284–288.

Du, K.-L. (2010). 'Clustering: A neural network approach'. *Neural networks : the official journal of the International Neural Network Society* 23 (1), pp. 89–107.

Duangsoithong, R. & Windeatt, T. (2010). 'Correlation-Based and Causal Feature Selection Analysis for Ensemble Classifiers'. In: *Artificial Neural Networks in Pattern Recognition*. Cairo, Egypt, pp. 25–36.

Dumouchel, P., Dehak, N., Attabi, Y., Dehak, R. & Boufaden, N. (2009). 'Cepstral and long-term features for emotion recognition'. In: *Proceedings of the Interspeech 2009*. Brighton, UK, pp. 344–347.

Edu, J. S., Such, J. M. & Suarez-Tangil, G. (2019). 'Smart Home Personal Assistants: A Security and Privacy Review'. *ACM Computing Surveys (CSUR)* 53 (6), pp. 1–36.

Ekman, P. (2005). 'Basic Emotions'. In: *Handbook of Cognition and Emotion.* Hoboken, NY, USA: John Wiley & Sons, pp. 45–60.

Elman, J. L. (1990). 'Finding Structure in Time'. *Cognitive Science* 14.2, pp. 179–211.

Elsholz, J.-P., de Melo, G., Hermann, M. & Weber, M. (2009). 'Designing an extensible architecture for Personalized Ambient Information'. *Pervasive and Mobile Computing* 5.5, pp. 592–605.

Emmert-Streib, F., Yang, Z., Feng, H., Tripathi, S. & Dehmer, M. (2020). 'An Introductory Review of Deep Learning for Prediction Models With Big Data'. *Frontiers in Artificial Intelligence* 3, s.p.

Ephratt, M. (2011). 'Linguistic, paralinguistic and extralinguistic speech and silence'. *Journal of Pragmatics* 43.9, pp. 2286–2307.

Espinosa, H., Martínez-Miranda, J., Espinosa Curiel, I., Rodríguez-Jacobo, J. & Avila-George, H. (2017). 'Using acoustic paralinguistic information to assess the interaction quality in speech-based systems for elderly users'. *International Journal of Human-Computer Studies* 98, pp. 1–13.

Eyben, F., Wöllmer, M. & Schuller, B. (2010). 'openSMILE - The Munich Versatile and Fast Open-Source Audio Feature Extractor'. In: *Proceedings of the ACM Multimedia 2010 International Conference*, pp. 1459–1462.

Fragopanagos, N. & Taylor, J. (2005). 'Emotion recognition in human–computer interaction'. *Neural Networks* 18.4, pp. 389–405.

French, M. & Handy, R. (2007). 'Spectrograms: Turning Signals into Pictures'. *Journal of Engineering Technology* 24 (1), pp. 32–35.

Ghahramani, Z. (2004). 'Unsupervised Learning'. In: *Advanced Lectures on Machine Learning: ML Summer Schools 2003.* Canberra, Australia / Tübingen, Germany, pp. 72–112.

Girden, E. R. (1992). *ANOVA: Repeated measures.* 84. Newbury Park, CA, USA: Sage Publications, Inc.

Glüge, S. (2013). ‘Implicit Sequence Learning in Recurrent Neural Networks’. PhD thesis. Otto von Guericke University Magdeburg.

Gong, C. (2009). ‘Human-Machine Interface: Design Principles of Visual Information in Human-Machine Interface Design’. In: *2009 International Conference on Intelligent Human-Machine Systems and Cybernetics*. Vol. 2. Hangzhou, China, pp. 262–265.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. (2020). ‘Generative adversarial networks’. *Communications of the ACM* 63.11, pp. 139–144.

Grosan, C. & Abraham, A. (2011). ‘Rule-Based Expert Systems’. In: *Intelligent Systems: A Modern Approach*. Berlin, Germany: Springer, pp. 149–185.

Gross, J. J., Carstensen, L. L., Pasupathi, M., Tsai, J., Skorpen, C. G. & Hsu, A. Y. (1997). ‘Emotion and aging: experience, expression, and control.’ *Psychological Aging* 12 (4), pp. 590–599.

Grünenfelder, J., Zierau, N. & Janson, A. (2021). ‘Alexa, are you still there? Understanding the Habitual Use of AI-Based Voice Assistants’. In: *International Conference on Information Systems (ICIS 2021)*. Austin, TX, USA, p. 2066.

Guhr, N., Werth, O., Blacha, P. P. H. & Breitner, M. H. (2020). ‘Privacy concerns in the smart home context’. *SN Applied Sciences* 2.2, p. 247.

Hansen, J. H. L. & Bou-Ghazale, S. E. (1997). ‘Getting started with SUSAS: a speech under simulated and actual stress database’. *5th European Conference on Speech Communication and Technology (Eurospeech 1997)*, pp. 1743–1746.

Harrington, J., Palethorpe, S. & Watson, C. I. (2007). ‘Age-related changes in fundamental frequency and formants : a longitudinal study of four speakers’. In: *Proceedings of the Interspeech 2007*. Vol. 2. Antwerp, Belgium, pp. 1081–1084.

Hayashi, V. & Ruggiero, W. (2020). 'Non-Invasive Challenge Response Authentication for Voice Transactions with Smart Home Behavior'. *Sensors* 20.22, p. 6563.

Hochreiter, S. & Schmidhuber, J. (1997). 'Long Short-Term Memory'. *Neural computation* 9.8, pp. 1735–1780.

Hoffmann, H., Scheck, A., Schuster, T., Walter, S., Limbrecht, K., Traue, H. C. & Kessler, H. (2012). 'Mapping discrete emotions into the dimensional space: An empirical approach'. In: *2012 IEEE International Conference on Systems, Man, and Cybernetics*. Seoul, Korea, pp. 3316–3320.

Honold, F. et al. (2014). 'Companion-Technology: Towards User- and Situation-Adaptive Functionality of Technical Systems'. In: *2014 International Conference on Intelligent Environments*. Shanghai, China, pp. 378–381.

Huang, X., Ariki, Y. & Jack, M. (1990). *Hidden Markov Models for Speech Recognition*. New York, NY, USA: Columbia University Press.

Islam, M. A., Jia, S. & Bruce, N. D. B. (2020). 'How Much Position Information Do Convolutional Neural Networks Encode?' *International Conference on Learning Representations (ICLR 2020)*, s.p.

Izhar, F., Ali, S., Ponum, M., Mahmood, M. T., Ilyas, H. & Iqbal, A. (2023). 'Detection & recognition of veiled and unveiled human face on the basis of eyes using transfer learning'. *Multimedia Tools and Applications* 82.3, pp. 4257–4287.

Jakobson, G., Corp, A., Parameswaran, N., Buford, J., Lewis, L. & Ray, P. (2006). 'Situation-Aware Multi-Agent System for Disaster Relief Operations Management'. *Environmental Science, Engineering, Computer Science* 4, pp. 1–8.

Janiesch, C., Zschech, P. & Heinrich, K. (2021). 'Machine learning and deep learning'. *Electronic Markets* 31.3, pp. 685–695.

Jat, A. & Grønli, T.-M. (2022). 'Smart Watch for Smart Health Monitoring: A Literature Review'. In: *Bioinformatics and Biomedical Engineering*. Cham, Switzerland: Springer International Publishing, pp. 256–268.

Jürgens, R., Hammerschmidt, K. & Fischer, J. (2011). 'Authentic and Play-Acted Vocal Emotion Expressions Reveal Acoustic Differences'. *Frontiers in Psychology* 2, p. 180.

Kameas, A. D., Goumopoulos, C., Hagras, H., Callaghan, V., Heinroth, T. & Weber, M. (2009). 'An Architecture That Supports Task-Centered Adaptation In Intelligent Environments'. In: *Advanced Intelligent Environments*. Boston, MA, USA: Springer, pp. 41–66.

Kappas, A., Hess, U. & Scherer, K. (1991). 'Voice and Emotion'. In: *Fundamentals of Nonverbal Behavior*. Cambridge, UK: Cambridge University Press, pp. 200–234.

Karray, F., Alemzadeh, M., Saleh, J. & Arab, M. N. (2008). 'Human-Computer Interaction: Overview on State of the Art'. *International Journal on Smart Sensing and Intelligent Systems* 1 (1), pp. 137–159.

Kawaguchi, K., Kaelbling, L. P. & Bengio, Y. (2017). 'Generalization in Deep Learning'. *Mathematical Aspects of Deep Learning*, pp. 112–148.

Kaya, H., Özkaptan, T., Salah, A. & Gurgen, F. (2014). 'Canonical Correlation Analysis and Local Fisher Discriminant Analysis based Multi-View Acoustic Feature Reduction for Physical Load Prediction'. In: *Proceedings of the Interspeech 2014*. Singapore, Singapore, pp. 442–446.

Kelley, J. F. (1984). 'An iterative design methodology for user-friendly natural language office information applications'. *ACM Transactions on Information Systems (TOIS)* 2.1, pp. 26–41.

Kindsvater, D., Meudt, S. & Schwenker, F. (2017). 'Fusion Architectures for Multimodal Cognitive Load Recognition'. In: *Multimodal Pattern Recognition of Social Signals in Human-Computer-Interaction*. Cancun, Mexico, pp. 36–47.

Knight, E., Hernandez, S., Bayne, E., Bulitko, V. & Tucker, B. (2019). 'Pre-processing spectrogram parameters improve the accuracy of bioacoustic classification using convolutional neural networks'. *Bioacoustics* 29 (3), pp. 1–19.

Kohl, L., Eschenbacher, S., Besinger, P. & Ansari, F. (2024). 'Large Language Model-based Chatbot for Improving Human-Centricity in Maintenance Planning and Operations'. In: *PHM Society European Conference*. Vol. 8. Prague, Czech Republic, p. 12.

Kotseruba, I. & Tsotsos, J. K. (2020). '40 years of cognitive architectures: core cognitive abilities and practical applications'. *Artificial Intelligence Review* 53.1, pp. 17–94.

Koza, J. R., Bennett, F. H., Andre, D. & Keane, M. A. (1996). 'Automated Design of Both the Topology and Sizing of Analog Electrical Circuits Using Genetic Programming'. In: *Artificial Intelligence in Design '96*. Dordrecht, The Netherlands: Springer Netherlands, pp. 151–170.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2017). 'ImageNet Classification with Deep Convolutional Neural Networks'. In: *Communications of the ACM*. Vol. 60, pp. 84–90.

Kruskal, W. H. & Wallis, W. A. (1952). 'Use of Ranks in One-Criterion Variance Analysis'. *Journal of the American Statistical Association* 47.260, pp. 583–621.

Kumar, N. & Lee, S. C. (2022). 'Human-machine interface in smart factory: A systematic literature review'. *Technological Forecasting and Social Change* 174 (25), pp. 121–284.

Laird, J. E. (2012). *The Soar Cognitive Architecture*. Cambridge, MA, USA: The MIT Press.

Lee, J. (2015). 'Smart Factory Systems'. *Informatik-Spektrum* 38 (3), pp. 230–235.

Lee, K., Kay, J., Kang, B. & Rosebrock, U. (2002). 'A Comparative Study on Statistical Machine Learning Algorithms and Thresholding Strategies for Automatic Text Categorization'. In: *Proceedings of the 7th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence*. Vol. 2417. Tokyo, Japan, pp. 444–453.

Lew, M., Bakker, E. M., Sebe, N. & Huang, T. S. (2007). 'Human-Computer Intelligent Interaction: A Survey'. In: *Human–Computer Interaction (HCI 2007).* Rio de Janeiro, Brazi, pp. 1–5.

Li, J., Cheng, J.-h., Shi, J.-y. & Huang, F. (2012). 'Brief Introduction of Back Propagation (BP) Neural Network Algorithm and Its Improvement'. *Advances in Intelligent and Soft Computing* 169, pp. 553–558.

Liptak, A. (2017). 'Amazon's Alexa started ordering people dollhouses after hearing its name on TV'. [Online; posted 07-Jan-2017], https://perma.cc/CQA9-R3UA.

Liz Dean, B. C. & Ruppanner, L. (2022). 'The mental load: building a deeper theoretical understanding of how cognitive and emotional labor overload women and mothers'. *Community, Work & Family* 25.1, pp. 13–29.

Lopatovska, I., Rink, K., Knight, I., Raines, K., Cosenza, K., Williams, H., Sorsche, P., Hirsch, D., Li, Q. & Martinez, A. (2018). 'Talk to me: Exploring user interactions with the Amazon Alexa'. *Journal of Librarianship and Information Science* 51 (4), pp. 984–997.

Lotz, A., Siegert, I. & Wendemuth, A. (2016). 'Comparison of Different Modeling Techniques for Robust Prototype Matching of Speech Pitch-Contours'. *Kognitive Systeme* 2016.1, s.p.

Luo, H., Koszalka, T. & Zuo, M. (2016). 'Investigating the Effects of Visual Cues in Multimedia Instruction Using Eye Tracking'. In: *International Conference on Blended Learning.* Vol. 9757, pp. 63–72.

Mache, S. R., Baheti, M. R. & Mahender, C. N. (2015). 'Review on Text-To-Speech Synthesizer'. *International Journal of Advanced Research in Computer and Communication Engineering* 4, pp. 54–59.

Malinowski, A., Cholewo, T. & Zurada, J. (1995). 'Capabilities and limitations of feedforward neural networks with multilevel neurons'. In: *IEEE International Symposium on Circuits and Systems (ISCAS 1995).* Vol. 1, pp. 131–134.

Mallidi, S. H., Maas, R., Goehner, K., Rastrow, A., Matsoukas, S. & Hoffmeister, B. (2018). 'Device-directed Utterance Detection'. In: *Proceedings of the Interspeech 2018*. Hyderabad, India, pp. 1225–1228.

Martin, O., Kotsia, I., Macq, B. & Pitas, I. (2006). 'The eNTERFACE'05 Audio-Visual Emotion Database'. In: *22nd International Conference on Data Engineering Workshops (ICDEW 2006)*. Atlanta, GA, USA, p. 8.

Masmoudi, O., Jaoua, M., Jaoua, A. & Yacout, S. (2021). 'Data Preparation in Machine Learning for Condition-based Maintenance'. *Journal of Computer Science* 17 (6), pp. 525–538.

Mauss, I. & Robinson, M. (2009). 'Measures of emotion: A review'. *Cognition & Emotion* 23 (2), pp. 209–237.

McCulloch, W. S. & Pitts, W. (1943). 'A logical calculus of the ideas immanent in nervous activity'. *The bulletin of mathematical biophysics* 5.4, pp. 115–133.

Melo, C. M. de, Kim, K., Norouzi, N., Bruder, G. & Welch, G. (2020). 'Reducing Cognitive Load and Improving Warfighter Problem Solving With Intelligent Virtual Assistants'. *Frontiers in Psychology* 11, s.p.

Merten, M., Bley, A., Schroeter, C. & Gross, H.-M. (2012). 'A mobile robot platform for socially assistive home-care applications'. In: *7th German Conference on Robotics (ROBOTIK 2012)*. Munich, Germany, pp. 1–6.

Meurisch, C., Mihale-Wilson, C. A., Hawlitschek, A., Giger, F., Müller, F., Hinz, O. & Mühlhäuser, M. (2020). 'Exploring User Expectations of Proactive AI Systems'. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies,* 4.4, pp. 1–22.

Møller, M. (1993). 'A Scaled Conjugate Gradient Algorithm For Fast Supervised Learning'. *Neural Networks* 6 (4), pp. 525–533.

Moors, A., Ellsworth, P., Scherer, K. & Frijda, N. (2013). 'Appraisal Theories of Emotion: State of the Art and Future Development'. *Emotion Review* 5 (2), pp. 119–124.

Müller, V. C. (2011). 'Interaction and Resistance: The Recognition of Intentions in New Human-Computer Interaction'. In: *Toward Autonomous, Adaptive,* *and Context-Aware Multimodal Interfaces. Theoretical and Practical Issues.* Vol. 6456, pp. 1–7.

Murshed, M. G. S., Murphy, C., Hou, D., Khan, N., Ananthanarayanan, G. & Hussain, F. (2019). 'Machine Learning at the Network Edge: A Survey'. *ACM Computing Surveys* 54 (8), pp. 1–37.

Nassif, A. B., Shahin, I., Attili, I., Azzeh, M. & Shaalan, K. (2019). 'Speech Recognition Using Deep Neural Networks: A Systematic Review'. *IEEE Access* 7, pp. 19143–19165.

Natsiou, A. & O'Leary, S. (2021). 'Audio representations for deep learning in sound synthesis: A review'. In: *18th International Conference on Computer Systems and Applications (AICCSA 2021).* Tangier, Morocco, pp. 1–8.

Nguyen, A., Oberföll, A. & Färber, M. (2020). 'Right for the Right Reason: Making Image Classification Robust'. *arXiv* abs/2007.11924v2, s.p.

Ötting, S. K., Masjutin, L., Steil, J. J. & Maier, G. W. (2020). 'Let's Work Together: A Meta-Analysis on Robot Design Features That Enable Successful Human–Robot Interaction at Work'. *Human Factors* 64 (6), pp. 1027–1050.

Pandey, R., Castillo, C. & Purohit, H. (2019). 'Modeling human annotation errors to design bias-aware systems for social stream processing'. In: *International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2019).* Vancouver, British Columbia, Canada, pp. 374–377.

Paul, S., Uddin, M. & Bouakaz, S. (2014). 'Face Recognition using Eyes, Nostrils and Mouth Features'. In: *16th International Conference on Computer and Information Technology.* Khulna, Bangladesh, pp. 117–120.

Paullada, A., Raji, I. D., Bender, E. M., Denton, E. & Hanna, A. (2021). 'Data and its (dis)contents: A survey of dataset development and use in machine learning research'. *Patterns* 2 (11), p. 100336.

Pearl, J. (1985). 'Bayesian Networks: A Model of Self-activated Memory for Evidential Reasoning'. In: vol. 850021. Report (University of California, Los Angeles. Computer Science Dept.) Los Angeles, CA, USA: UCLA Computer Science Department.

Pearson, K. (1901). 'LIII. On lines and planes of closest fit to systems of points in space'. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11, pp. 559–572.

Picard, R., Vyzas, E. & Healey, J. (2001). 'Toward Machine Emotional Intelligence: Analysis of Affective Physiological State'. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23 (10), pp. 1175–1191.

Picard, R. W. (2003). 'Affective Computing: Challenges'. *International Journal of Human-Computer Studies* 59.1, pp. 55–64.

Pisanski, K., Jones, B., Fink, B., O'Connor, J., DeBruine, L., Röder, S. & Feinberg, D. (2016). 'Voice parameters predict sex-specific body morphology in men and women'. *Animal Behaviour* 112, pp. 13–22.

Plutchik, R. (2001). 'The Nature of Emotions'. *American Scientist* 89.4, pp. 344–350.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., Shyu, M.-L., Chen, S.-C. & Iyengar, S. S. (2018). 'A Survey on Deep Learning: Algorithms, Techniques, and Applications'. *A Survey on Deep Learning: Algorithms, Techniques, and Applications* 51.5, pp. 1–36.

Powers, D. M. W. (2011). 'Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation'. *Journal of Machine Learning Technologies* 2, pp. 37–63.

Preece, J., Rogers, Y. & Sharp, H. (2015). *Interaction Design: Beyond Human-Computer Interaction*. 4th ed. Hoboken, NJ, USA: Wiley.

Pugachev, A., Akhtiamov, O., Karpov, A. & Minker, W. (2018). 'Deep Learning for Acoustic Addressee Detection in Spoken Dialogue Systems'. In: *Artificial Intelligence and Natural Language (AINL 2017)*. Saint Petersburg, Russia, pp. 45–53.

Qiu, J., Wu, Q., Ding, G., Xu, Y. & Feng, S. (2016). 'A survey of machine learning for big data processing'. *EURASIP Journal on Advances in Signal Processing* 2016.1, p. 67.

Rahman, M. M., Islam, M. S., Sassi, R. & Aktaruzzaman, M. (2019). 'Convolutional neural networks performance comparison for handwritten Bengali numerals recognition'. *SN Applied Sciences* 1.12, p. 1660.

Rakotomalala, F., Randriatsarafara, H., Hajalalaina, A. & Ndaohialy Manda Vy, R. (2021). 'Voice User Interface: Literature Review, Challenges and Future Directions'. *System Theory, Control And Computing Journal* 1.2, pp. 65–89.

Rao, A. S. & Georgeff, M. P. (1991). 'Modeling Rational Agents within a BDI-Architecture'. In: *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR 2019).* Cambridge MA, USA, pp. 473–484.

Reed, C. L., Moody, E. J., Mgrublian, K., Assaad, S., Schey, A. & McIntosh, D. N. (2020). 'Body Matters in Emotion: Restricted Body Movement and Posture Affect Expression and Recognition of Status-Related Emotions'. *Frontiers in Psychology* 11, s.p.

Rejeb, A., Rejeb, K., Simske, S. & Treiblmaier, H. (2021). 'Humanitarian Drones: A Review and Research Agenda'. *Internet of Things* 16, p. 100434.

Resseguier, B., Léger, P.-M., Sénécal, S., Bastarache-Roberge, M.-C. & Courtemanche, F. (2016). 'The Influence of Personality on Users' Emotional Reactions'. In: *Proceedings. of Third International Conference on the HCI in Business, Government, and Organizations: Information Systems.* Toronto, Canada, pp. 91–98.

Rezaei, M. & Sabzevari, R. (2009). 'Multisensor Data Fusion Strategies for Advanced Driver Assistance Systems'. In: *Sensor and Data Fusion.* Rijeka, Croatia: IntechOpen. Chap. 8.

Richey, C. et al. (2018). 'Voices Obscured in Complex Environmental Settings (VOICES) corpus'. In: *Proceedings of Interspeech 2018.* Hyderabad, India, pp. 1566–1570.

Ricquebourg, V., Menga, D., Durand, D., Marhic, B., Delahoche, L. & Logé, C. (2007). 'The Smart Home Concept : our immediate future'. In: *1st IEEE International Conference on E-Learning in Industrial Electronics (ICELIE 2006).* Hammamet, Tunisia, pp. 23–28.

Rock, L. Y., Tajudeen, F. P. & Chung, Y. W. (2022). 'Usage and impact of the internet-of-things-based smart home technology: a quality-of-life perspective'. *Universal Access in the Information Society* 20 (1), pp. 345–364.

Rosenfeld, A. & Richardson, A. (2019). 'Explainability in Human-Agent Systems'. *Autonomous Agents and Multi-Agent Systems* 33.6.

Rublee, E., Rabaud, V., Konolige, K. & Bradski, G. (2011). 'ORB: an efficient alternative to SIFT or SURF'. In: *IEEE International Conference on Computer Vision (ICCV 2011).* Barcelona, Spain, pp. 2564–2571.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986). 'Learning representations by back-propagating errors'. *Nature* 323, pp. 533–536.

Russell, J. (1980). 'A Circumplex Model of Affect'. *Journal of Personality and Social Psychology* 39 (6), pp. 1161–1178.

Russo, S. et al. (2021). 'The value of human data annotation for machine learning based anomaly detection in environmental systems'. *Water Research* 206 (7), p. 117695.

Sakr, G. E., Mokbel, M., Darwich, A., Khneisser, M. N. & Hadi, A. (2016). 'Comparing deep learning and support vector machines for autonomous waste sorting'. In: *IEEE International Multidisciplinary Conference on Engineering Technology (IMCET 2016).* Beirut, Lebanon, pp. 207–212.

Salkuti, S. R. (2020). 'A survey of big data and machine learning'. *International Journal of Electrical and Computer Engineering (IJECE)* 10.1, pp. 575–580.

Sapra, A., Panwar, N. & Panwar, S. (2013). 'Emotion Recognition from Speech'. *International Journal of Emerging Technology and Advanced Engineering* 3.2, pp. 341–345.

Sarker, I. H. (2021a). 'Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions'. *SN Computer Science* 2.6, p. 420.

— (2021b). 'Machine Learning: Algorithms, Real-World Applications and Research Directions'. *SN Computer Science* 2.3, p. 160.

Scherer, K. (2005). 'What are emotions? And how can they be measured?' *Social Science Information* 44.4, pp. 695–729.

Schmidhuber, J. (2015). 'Deep learning in neural networks: An overview'. *Neural Networks* 61, pp. 85–117.

Schuller, B., Steidl, S., Batliner, A., Burkhardt, F., Devillers, L., Mueller, C. & Narayanan, S. (2010). 'The INTERSPEECH 2010 Paralinguistic Challenge'. In: *Proceedings of the Interspeech 2010*. Makuhari, Japan, pp. 2794–2797.

Schuller, B., Steidl, S., Batliner, A., Schiel, F. & Krajewski, J. (2011). 'The INTERSPEECH 2011 Speaker State Challenge'. In: *Proceedings of the Interspeech 2011*. Florence, Italy, pp. 3201–3204.

Schuller, B., Vlasenko, B., Eyben, F., Rigoll, G. & Wendemuth, A. (2009). 'Acoustic Emotion Recognition: A Benchmark Comparison of Performances'. In: *Automatic Speech Recognition & Understanding (ASRU 2009)*. Moreno, Italy, pp. 552–557.

Schuller, B., Friedmann, F. & Eyben, F. (2014a). 'The Munich Biovoice Corpus: Effects of Physical Exercising, Heart Rate, and Skin Conductance on Human Speech Production'. In: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*. Reykjavik, Iceland, pp. 1506–1510.

Schuller, B., Rigoll, G. & Lang, M. (2004). 'Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine-belief network architecture'. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004)*. Vol. 1. Montreal, QC, Canada, pp. I–577.

Schuller, B., Steidl, S., Batliner, A., Hirschberg, J., Burgoon, J., Baird, A., Elkins, A., Zhang, Y., Coutinho, E. & Evanini, K. (2016). 'The INTERSPEECH 2016 Computational Paralinguistics Challenge: Deception, Sincerity and Native Language'. *Proceedings of the Interspeech 2016*, pp. 2001–2005.

Schuller, B., Steidl, S., Batliner, A., Krajewski, J., Epps, J., Eyben, F., Ringeval, F., Marchi, E. & Schnieder, S. (2014b). 'The Interspeech 2014 computational paralinguistics challenge: Cognitive & physical load'. In: s.p.

Sevilla, J., Heim, L., Ho, A., Besiroglu, T., Hobbhahn, M. & Villalobos, P. (2022). 'Compute Trends Across Three Eras of Machine Learning'. In: *International Joint Conference on Neural Networks (IJCNN 2022).* Padua, Italy, pp. 1–18.

Sezgin, M., Gunsel, B. & Kurt, G. (2012). 'Perceptual audio features for emotion detection'. *EURASIP Journal on Audio, Speech, and Music Processing* 2012 (1), p. 16.

Shadeed, I., Abd, D., Alwan, J. & Rabash, A. (2018). 'Performance Evaluation of Kernels in Support Vector Machine'. In: *1st Annual International Conference on Information and Sciences (AiCIS 2018).* Fallujah, Iraq, pp. 96–101.

Shriberg, E., Stolcke, A., Hakkani-Tür, D. & Heck, L. (2012). 'Learning When to Listen: Detecting System-Addressed Speech in Human-Human-Computer Dialog'. In: *Proceedings of the Interspeech 2012.* Vol. 1. Portland, OR, USA, pp. 334–337.

Shriberg, E., Stolcke, A. & Ravuri, S. (2013). 'Addressee Detection for Dialog Systems Using Temporal and Spectral Dimensions of Speaking Style'. In: *Proceedings of the Interspeech 2013.* Lyon, France, pp. 2559–2563.

Sidorov, M., Schmitt, A., Semenkin, E. & Minker, W. (2016). 'Could Speaker, Gender or Age Awareness be beneficial in Speech-based Emotion Recognition?' In: *Proceedings of the Tenth LREC.* Portorož, Slovenia, pp. 61–68.

Siegert, I., Nietzold, J., Heinemann, R. & Wendemuth, A. (2019). 'The Restaurant Booking Corpus - content-identical comparative human-human and human-computer simulated telephone conversations'. In: *Elektronische Sprachsignalverarbeitung 2019. Tagungsband der 30. Konferenz.* Dresden, Germany, pp. 126–133.

Siegert, I., Glodek, M., Panning, A., Krell, G., Schwenker, F., Al-Hamadi, A. & Wendemuth, A. (2013). 'Using speaker group dependent modelling to improve fusion of fragmentary classifier decisions'. In: *IEEE International*

*Conference on Cybernetics (CYBCO 2013)*. Lausanne, Switzerland, pp. 132–137.

Siegert, I. & Krüger, J. (2021). In: *Advances in Data Science: Methodologies and Applications*. Vol. 189. Cham, Switzerland: Springer International Publishing. Chap. "Speech Melody and Speech Content Didn't Fit Together"—Differences in Speech Behavior for Device Directed and Human Directed Interactions, pp. 65–95.

Siegert, I., Philippou-Hübner, D., Hartmann, K., Böck, R. & Wendemuth, A. (2014). 'Investigation of Speaker Group-Dependent Modelling for Recognition of Affective States from Speech'. *Cognitive Computation* 6.4, pp. 892–913.

Siegert, I., Shuran, T. & Lotz, A. F. (2018). 'Acoustic Addressee-Detection – Analysing the Impact of Age, Gender and Technical Knowledge'. In: *Elektronische Sprachsignalverarbeitung 2018. Tagungsband der 29. Konferenz*. Ulm, Germany, pp. 118–125.

Siegert, I., Sinha, Y., Winkelmann, G., Jokisch, O. & Wendemuth, A. (2022a). 'Public Interactions with Voice Assistant – Discussion of Different One-Shot Solutions to Preserve Speaker Privacy'. In: *Proceedings of the Workshop on Ethical and Legal Issues in Human Language Technologies and Multilingual De-Identification of Sensitive Data*. Marseille, France, pp. 44–47.

Spratt, E. L. (2018). 'Dream Formulations and Deep Neural Networks: Humanistic Themes in the Iconology of the Machine-Learned Image'. *arXiv* abs/1802.01274, s.p.

Steininger, S., Rabold, S., Dioubina, O. & Schiel, F. (2002). 'Development of the User-State Conventions for the Multimodal Corpus in SmartKom'. In: *Proceedings of the 3rd International Conference on Language Resources and Evaluation, Workshop in Multimodal Resources and Multimodal Systems Evaluation*, pp. 33–37.

Strasser, A. (2022). 'Distributed responsibility in human–machine interactions'. *AI and Ethics* 2.3, pp. 523–532.

Stuhlsatz, A., Meyer, C., Eyben, F., Zielke, T., Meier, H.-G. & Schuller, B. W. (2011). 'Deep neural networks for acoustic emotion recognition: Raising the benchmarks.' In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2011).* Prague, Czech Republic, pp. 5688–5691.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. & Rabinovich, A. (2014a). 'Going Deeper with Convolutions'. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015).* Boston, MA, USA, pp. 1–9.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. & Fergus, R. (2014b). 'Intriguing properties of neural networks'. In: *2nd International Conference on Learning Representations (ICLR 2014).* Banff, Canada, s.p.

Tahon, M. & Devillers, L. (2016). 'Towards a Small Set of Robust Acoustic Features for Emotion Recognition: Challenges'. *IEEE/ACM Transactions Audio, Speech and Language Processing* 24.1, pp. 16–28.

Taylor, S., Dromey, C., Nissen, S. L., Tanner, K., Eggett, D. & Corbin-Lewis, K. (2020). 'Age-Related Changes in Speech and Voice: Spectral and Cepstral Measures'. *Journal of speech, language, and hearing research : JSLHR* 63.3, pp. 647–660.

Tong, X., Huang, C.-W., Mallidi, S. H., Joseph, S., Pareek, S., Chandak, C., Rastrow, A. & Maas, R. (2021). 'Streaming ResLSTM with Causal Mean Aggregation for Device-Directed Utterance Detection'. In: *IEEE Spoken Language Technology Workshop (SLT 2021).* Shenzhen, China, pp. 659–664.

Tornow, M., Krippl, M., Bade, S., Thiers, A., Siegert, I., Handrich, S., Krüger, J., Schega, L. & Wendemuth, A. (2016). 'Integrated Health and Fitness (iGF)-Corpus - ten-Modal Highly Synchronized Subject-Dispositional and Emotional Human Machine Interactions'. In: *Multimodal Corpora: Computer vision and language processing (MMC 2016).* Portorož, Slovenia, s.p.

Traum, D., Aggarwal, P., Artstein, R., Foutz, S., Gerten, J., Katsamanis, A., Leuski, A., Noren, D. & Swartout, W. (2012). 'Ada and Grace: Direct Interaction with Museum Visitors'. In: *Proceedings of the 12th international* *conference on Intelligent Virtual Agents*. Santa Cruz, CA, USA, pp. 245–251.

Truong, K., Nieuwenhuys, A., Beek, P. & Evers, V. (2015). 'A database for analysis of speech under physical stress: detection of exercise intensity while running and talking'. In: *Proceedings of the Interspeech 2015*. Dresden, Germany, pp. 3705–3709.

Tsai, J., Stolcke, A. & Slaney, M. (2015a). 'A Study of Multimodal Addressee Detection in Human-Human-Computer Interaction'. *IEEE Transactions on Multimedia* 17.9, pp. 1550–1561.

— (2015b). 'Multimodal addressee detection in multiparty dialogue systems'. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015)*. South Brisbane, Australia, pp. 2314–2318.

Valli, A. (2008). 'The design of natural interaction'. *Multimedia Tools and Applications* 38 (3), pp. 295–305.

Ververidis, D. & Kotropoulos, C. (2006). 'Emotional speech recognition: Resources, features, and methods'. *Speech Communication* 48 (9), pp. 1162–1181.

Vinciarelli, A., Pantic, M. & Bourlard, H. (2009). 'Social Signal Processing: Survey of an Emerging Domain'. *Image and Vision Computing* 27 (12), pp. 1743–1759.

Vinyals, O., Bohus, D. & Caruana, R. (2012). 'Learning Speaker, Addressee and Overlap Detection Models from Multimodal Streams'. In: *Proceedings of the 14th ACM International Conference on Multimodal Interaction*. Santa Monica, CA, USA, pp. 417–424.

Vlasenko, B., Schuller, B., Tadesse Mengistu, K., Rigoll, G. & Wendemuth, A. (2008). 'Balancing Spoken Content Adaptation and Unit Length in the Recognition of Emotion and Interest'. In: *Proceedings of the Interspeech 2008*. Brisbane, Australia, pp. 805–808.

Vogt, T. & André, E. (2005). 'Comparing Feature Sets for Acted and Spontaneous Speech in View of Automatic Emotion Recognition'. In: *IEEE International*

*Conference on Multimedia and Expo 2005.* Amsterdam, The Netherlands, pp. 474–477.

Wahlster, W. (2006). *SmartKom: Foundations of Multimodal Dialogue Systems.* Cognitive Technologies. Berlin, Germany: Springer.

Wallingford, M., Li, H., Achille, A., Ravichandran, A., Fowlkes, C., Bhotika, R. & Soatto, S. (2022). 'Task Adaptive Parameter Sharing for Multi-Task Learning'. In: *Conference on Computer Vision and Pattern Recognition (CVPR 2022).* New Orleans, LA, USA, pp. 7551–7560.

Weiss, K., Khoshgoftaar, T. M. & Wang, D. (2016). 'A survey of transfer learning'. *Journal of Big Data* 3.1, p. 9.

Wienrich, C., Reitelbach, C. & Carolus, A. (2021). 'The Trustworthiness of Voice Assistants in the Context of Healthcare Investigating the Effect of Perceived Expertise on the Trustworthiness of Voice Assistants, Providers, Data Receivers, and Automatic Speech Recognition'. *Frontiers in Computer Science* 3, p. 685250.

Wilks, Y. (2005). 'Artificial companions'. *Interdisciplinary Science Reviews* 30, pp. 145–152.

Williams, R. J. & Zipser, D. (1995). 'Gradient-based learning algorithms for recurrent networks and their computational complexity'. In: *Back-propagation: Theory, Architectures and Applications.* Hillsdale, MI, USA: L. Erlbaum Associates Inc. Chap. 13, pp. 433–486.

Wöllmer, M., Eyben, F., Reiter, S., Schuller, B., Cox, C., Douglas-Cowie, E. & Cowie, R. (2008). 'Abandoning Emotion Classes – Towards Continuous Emotion Recognition with Modelling of Long-Range Dependencies'. In: *Proceedings of the Interspeech 2008.* Brisbane, Australia, pp. 597–600.

Wöllmer, M., Eyben, F., Schuller, B., Douglas-Cowie, E. & Cowie, R. (2009). 'Data-driven clustering in emotional space for affect recognition using discriminatively trained LSTM networks'. In: *Proceedings of the Interspeech 2009.* Brighton, UK, pp. 1595–1598.

Xing, Y., Shen, F., Luo, C. & Zhao, J. (2015). 'L3-SVM: a lifelong learning method for SVM'. In: *International Joint Conference on Neural Networks (IJCNN 2015).* Killarney, Ireland, pp. 1–8.

Xu, X., Li, Y., Xu, X., Wen, Z., Che, H., Liu, S. & Tao, J. (2014). 'Survey on discriminative feature selection for speech emotion recognition'. In: *The 9th International Symposium on Chinese Spoken Language Processing.* Singapore, Singapore, pp. 345–349.

Yang, J., Liu, T., Liu, Y. & Morgan, P. (2022). 'Review of Human-Machine Interaction Towards Industry 5.0: Human-Centric Smart Manufacturing'. In: *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference.* Vol. Volume 2: 42nd Computers and Information in Engineering Conference (CIE). St. Louis, MO, USA, V002T02A060.

Yellamraju, S. (2013). 'Design of Various Logic Gates in Neural Networks'. In: *Annual IEEE India Conference (INDICON 2013).* Mumbai, India, pp. 1–5.

Ying, X. (2019). 'An Overview of Overfitting and its Solutions'. *Journal of Physics: Conference Series* 1168 (2), p. 022022.

Yu, A. C. & Eng, J. (2020). 'One Algorithm May Not Fit All: How Selection Bias Affects Machine Learning Performance'. *RadioGraphics* 40.7, pp. 1932–1937.

Zeiler, M. D. & Fergus, R. (2014). 'Visualizing and Understanding Convolutional Networks'. In: *13th European Conference Computer Vision (ECCV 2014).* Zurich, Switzerland, pp. 818–833.

Zheng, C., Abd-Elrahman, A. & Whitaker, V. (2021). 'Remote Sensing and Machine Learning in Crop Phenotyping and Management, with an Emphasis on Applications in Strawberry Farming'. *Remote Sensing* 13.3, p. 531.

Zheng, F., Zhang, G. & Song, Z. (2001). 'Comparison of different implementations of MFCC'. *Journal of Computer Science and Technology* 16.6, pp. 582–589.