Robert Herms

Effective Speech Features for Cognitive Load Assessment:
Classification and Regression

Wissenschaftliche Schriftenreihe
Dissertationen der Medieninformatik
Band 9

Prof. Dr. Maximilian Eibl (Hrsg.)

Robert Herms

Effective Speech Features for Cognitive Load Assessment: Classification and Regression

TECHNISCHE UNIVERSITÄT
CHEMNITZ

Universitätsverlag Chemnitz
2019

Impressum

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Angaben sind im Internet über http://www.dnb.de abrufbar.

Titelgrafik: iStock.com/filo
Satz/Layout: Robert Herms

Technische Universität Chemnitz/Universitätsbibliothek
Universitätsverlag Chemnitz
09107 Chemnitz
https://www.tu-chemnitz.de/ub/univerlag

readbox unipress
in der readbox publishing GmbH
Am Hawerkamp 31
48155 Münster
http://unipress.readbox.net

ISSN 2195-2574 print - ISSN 2195-2582 online
ISBN 978-3-96100-087-6
http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa2-333464

Robert Herms

Effective Speech Features for Cognitive Load Assessment: Classification and Regression

Doctoral Thesis
submitted in fulfilment of the requirements for the degree of
Doktoringenieur (Dr.-Ing.)

Faculty of Computer Science,
Technische Universität Chemnitz, Germany

Dissertation Committee:
Prof. Dr. Maximilian Eibl (Technische Universität Chemnitz, Germany)
Prof. Dr. Günter Daniel Rey (Technische Universität Chemnitz, Germany)

Submitted on 21 September 2018
Defended on 18 December 2018

Abstract

The automatic recognition of cognitive load is a vital step towards the development of adaptive systems that are capable of providing the user with dynamic support in order to maintain the load experienced within an optimal range for maximum productivity. Speech contains a multitude of information and has been identified to be a potential modality to measure the user's cognitive load.

The focus of this thesis is on the effectiveness of speech features for automatic cognitive load assessment, with particular attention being paid to new perspectives of this research area. A new cognitive load database, called CoLoSS, is introduced containing speech recordings of users who performed a learning task. This data collection contrasts with existing cognitive load databases since learning tasks have not yet been employed and it provides continuous numerical labels in addition to the discrete load levels considered until now. The CoLoSS corpus, together with the CLSE database in which two variants of the Stroop test and a reading span task are employed, forms the basis for the evaluations. Various acoustic features from different categories including prosody, voice quality, and spectrum are investigated in terms of their relevance. Moreover, Teager energy parameters, which have proven highly successful in stress detection, are introduced for cognitive load assessment and it is demonstrated how automatic speech recognition technology can be used to extract potential indicators of the user's cognitive load. As a further contribution, three hand-crafted feature sets are proposed.

The suitability of the extracted features is systematically evaluated by recognition experiments with speaker-independent systems designed for three-class classification (low, medium, and high cognitive load). Various configurations in terms of combinations of features, filters for feature selection, feature normalisation methods, and model parameters are tested. To prove the generalisation ability of the proposed feature sets, cross-corpus experiments are carried out. Additionally, a novel approach to speech-based cognitive load modelling is introduced, whereby the load is represented as a continuous quantity and its prediction can thus be regarded as a regression problem. The evaluation of regression algorithms on the CoLoSS corpus reveals the advantages of using automatic feature subset selection.

Acknowledgments

The research associated with this thesis was realised as part of the research project ‘StayCentered – MACeLot’ funded by the Federal Ministry of Education and Research (BMBF) under the reference 16SV7260. Also, the research was supported within the scope of the Research Training Group ‘CrossWorlds’ funded by the German Research Foundation (DFG) under the reference GRK 1780/1.

I would like to express my sincere gratitude to my supervisor Prof. Dr. Maximilian Eibl for his excellent guidance and valuable suggestions throughout my scientific work at the Technische Universität Chemnitz. He gave me the opportunity to work on this thesis at the chair of Media Informatics and allowed me enough freedom to define and explore my own directions in research but also supported me in keeping a sharp eye on what is essential. Furthermore, I have been given the opportunity to attend project meetings and international conferences. I would also like to thank Prof. Dr. Günter Daniel Rey, head of the chair of Psychology of Learning with Digital Media, for his willingness and effort to review this thesis. His unfailing advice and feedback have encouraged me to constantly improve the quality of my work.

I am grateful for the great collaboration with Dr. Maria Wirzberger. Her expertise and wide range of knowledge on cognitive load in instructional scenarios and on statistical methods have benefited me a lot. I enjoyed working with her very much—together we have designed the tasks for the CoLoSS corpus and carried out the corresponding recordings. I would like to extend my thanks to my colleagues at the chair of Media Informatics for the warm atmosphere and constructive criticism. Thanks are owed particularly to Dr. Stefanie Müller for many valuable discussions. Robert Manthey deserves thanks for his technical support regarding the experiments conducted in this thesis. Furthermore, I would like to thank the students Mike Ebersbach and Robert Alig for their technical assistance and helpful ideas.

Finally, special thanks go to my parents and my partner Stefanie for their encouragement, love, and care.

Chemnitz, Robert Herms
January 2019

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

ACC	Accuracy
ACF	Autocorrelation Function
ARPA	Advanced Research Projects Agency
ASR	Automatic Speech Recognition
CC_P	Pearson's Correlation Coefficient
CC_S	Spearman's Correlation Coefficient
CFS	Correlation-based Feature Selection
CL	Cognitive Load
CLSE	Cognitive Load with Speech and EGG
CLT	Cognitive Load Theory
CN	Corpus Normalisation
CoLoSS	Cognitive Load by Speech and Performance Data in a Symbol-Digit Dual-Task
ComParE	Computational Paralinguistics Challenge
CPP	Cepstral Peak Prominence
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
EEG	Electroencephalography
Eff_{PT}	Primary Task Efficiency
Eff_{ST}	Secondary Task Efficiency
EGG	Electroglottograph
F_0	Fundamental Frequency
$F_{1\text{–}4}(f,b)$	Centre Frequency (f) and Bandwidth (b) of Formant 1–4
FFT	Fast Fourier Transform
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
HNR	Harmonics-to-Noise Ratio
IG	Information Gain
IPA	International Phonetic Alphabet
LFPC	Log-Frequency Power Coefficient
LOCOCV	Leave-One-Corpus-Out Cross-Validation
LOSOCV	Leave-One-Subject-Out Cross-Validation
LPC	Linear Predictive Coding

LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MFCC	Mel Frequency Cepstral Coefficient
MLP	Multilayer Perceptron
MSE	Mean Square Error
NFD-LFPC	Nonlinear Frequency Domain Log-Frequency Power Coefficient
NTD-LFPC	Nonlinear Time Domain Log-Frequency Power Coefficient
OSPAN	Operation Span
PN	Partition Normalisation
RMS	Root Mean Square
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
SAMPA	Speech Assessment Methods Phonetic Alphabet
SC	Spectral Centroid
SCN	Speaker-Cluster Normalisation
SMA	Simple Moving Average
SMO	Sequential Minimal Optimisation
SMOTE	Synthetic Minority Oversampling Technique
SN	Speaker Normalisation
SVM	Support Vector Machine
SVR	Support Vector Regression
TEO	Teager Energy Operator
TEO-CB-Auto-Env	Critical Band based TEO Autocorrelation Envelope Area
TN	Training Normalisation
UAR	Unweighted Average Recall
VAD	Voice Activity Detection
VRD	Verbal Response Duration
WA	Word Accuracy

Chapter 1

Introduction

The rapid technical progress of solutions designed to support industry and everyday life is accompanied by issues relating to human-machine interaction. This applies particularly to environments in which a vast amount of information needs to be integrated and processed. In high accuracy and complex working environments, such as air traffic control or emergency response centres, tasks are highly demanding in terms of monitoring, reasoning, and decision-making (Majumdar and Ochieng, 2002). In this respect, high operator performance is required, often under sub-optimal conditions in which competing tasks have to be fulfilled within a relatively short period of time. These requirements are, however, not limited to error-critical working environments—in modern society, the intuitive interaction with computerised devices plays a major role, for example, in education (Gikas and Grant, 2013), car driving (Svangren et al., 2017), entertainment (Vinayagamoorthy et al., 2016), and more.

Assisting users in completing their tasks more efficiently and enhancing their performance constitutes a multidimensional problem including causal and assessment factors. The relationship between such factors can be explained through a construct, which is commonly known as *cognitive load* (Paas and Van Merriënboer, 1994a). Broadly speaking, cognitive load refers to the amount of mental demands imposed on a user by a particular task. This psychological construct is based on models of the human working memory that assume limitations in information processing concerning capacity and time (Miyake and Shah, 1999). The automatic assessment of the user's cognitive load is a vital step towards the development of adaptive interactive systems, which are capable of providing the user with dynamic support. In this regard, the vision is that, by adjusting the characteristics of tasks based on information about the user state, the load experienced can be maintained within an optimal range for maximum productivity and work safety.

1.1 Cognitive Load and the Role of Speech

Cognitive load can be regarded as a variable that attempts to explain and quantify human performance, learning, and cognitive processes. There are, however, a number of overlapping and sometimes divergent definitions of the construct. In the human factors psychological domain, a popular concept is the so-called 'mental load' or 'mental workload'. A good starting point to the issues relevant in the study and measurement of mental workload is provided by Moray (1979). The attributes of operator workload are broadly divided into input load, operator effort, and performance (Johannsen, 1979, p. 4). Different types of operator effort may occur according to different functions of human information processing, for example, perception effort, decision effort, and communication effort (Johannsen, 1979, p. 8). Sometimes, the term 'stress' instead of 'peak load' is used with nearly the same meaning (Johannsen, 1979, p. 5). Apart from the human factors domain, in neuroeconomics-focused research, the term 'cognitive effort' is associated with decisions about whether to engage and also about the 'intensity of engagement' (Westbrook and Braver, 2015). For the purpose of this thesis, the term 'cognitive load' is adopted along with a widely accepted definition—which rather relates to human learning—that it is a '... multidimensional construct that represents the load that performing a particular task imposes on the cognitive system ...' (Paas and Van Merriënboer, 1994a).

Due to the potential for many applications, cognitive load measurement has been an active research area in the last couple of decades. A variety of techniques, ranging from simple approaches such as questionnaires to highly involved procedures such as functional brain imaging, have been proposed over the years (Wierwille and Eggemeier, 1993; Just et al., 2003). Generally, indicators of cognitive load can be divided into performance measures, subjective ratings, physiological measures, and behavioural measures (Sweller et al., 1998; Brunken et al., 2003).

Among behavioural measures, speech-based methods have gained interest and, in fact, it has been found that speech is affected by cognitive load (cf. Berthold and Jameson, 1999; Müller et al., 2001; Keränen et al., 2004). Speech is a natural way of communication for human beings and contains a multitude of information that can be measured permanently in a contact-free way—the recording process is generally imperceptible for the monitored person. The speech signal is attractive in many real-life tasks, for example, telephone conversations, voice control systems, communication training, and language learning, and the speech material can be easily collected in inexpensive ways. Moreover, it has been demonstrated that a large amount of acoustic descriptors can be extracted and processed in real-time (cf. Eyben, 2015). Hence, the speech signal constitutes a promising source for cognitive load monitoring based on sophisticated audio analysis systems. The

utilisation of multimodality can improve the quality of monitoring systems since single continuous measures typically suffer from limitations such as noise or interruptions—one can easily integrate audio analysis into multimodel systems to enhance the robustness of cognitive load assessment (Chen et al., 2016).

In light of audio analysis, one key challenge is to find those speech parameters that correlate well with factors of cognitive load. In other words, *effective speech features* have to be determined, which is the main focus of this thesis. From a technical point of view, audio analysis systems are usually structured into two main stages: feature extraction and interpretation. The term 'feature' is used as a synonym for the term 'parameter' or 'variable'. Feature extraction serves two purposes (Lerch, 2012, pp. 4–5):

- *Dimensionality reduction*: The raw amount of recorded audio data is too large to handle it for further processing. For instance, one channel of a digital audio file in Compact Disc quality (44.1 kHz, 16 bits per sample) with a length of 5 minutes results in 25.2 MB. Instead, a series of features can be used to represent this data with considerably fewer values by suppressing irrelevant information.
- *Meaningful representation*: All information that can possibly be extracted are contained in the raw audio data. In order to focus on relevant aspects of the signal, it is necessary to transform the audio data into representations that can be interpreted by humans or machines.

Features can be categorised as low-level or high-level. Low-level features are generally considered not to be directly interpretable by humans, while high-level features represent information with a direct meaning. However, there is no clear definition for an objective distinction between these terms—it is often a context-dependent decision. The second stage of audio analysis systems uses the extracted features to interpret the information. The output can be a feature on a higher level (e.g., phonemes of a speech recogniser) used for another system or it can be the final result such as the level of cognitive load experienced by the speaker. This stage usually implies the utilisation of a model, which has been trained using a set of features extracted from a dataset and, in the case of supervised machine learning, the corresponding labels. The term 'label' refers to the target value, typically the meaning behind an observation. Each observation, often denoted as 'instance' in machine learning, is represented by the extracted features in form of a feature vector. A classification model assigns discrete labels to the unknown test instances, whereas the output of a regression model is a continuous numerical value.

1.2 Past Deficiencies

Over the last three decades, a body of work has been carried out discussing the influence of cognitive load on human speech production. A comparatively young field is the automatic recognition of cognitive load from speech—first attempts have been made by Yin et al. (2007). The authors developed a system, which extracts frame-based acoustic features and discriminates between three levels of cognitive load using Gaussian mixture models. In subsequent years, the suitability of various speech parameters was investigated for cognitive load classification (e.g., Boril et al., 2010; Le, 2012; Yap, 2012; Quatieri et al., 2015). Furthermore, an international research competition has been held at the INTERSPEECH 2014 conference with the goal to compare state-of-the-art systems of this topic area (cf. Schuller et al., 2014).

From the brief overview given above, it can be concluded that there exists great interest in the scientific community regarding optimal feature sets and machine learning algorithms for speech-based cognitive load recognition. However, not all aspects have been addressed in the past—there is still a gap in terms of use-case scenarios. Frequently encountered task designs include Stroop test with time pressure (Le et al., 2010a), reading comprehension (Le et al., 2011), driving under cognitive load (Boril et al., 2010), and arithmetic tasks (Gorovoy et al., 2010). These tasks are highly suitable to investigate the limitations of the human working memory but do not necessarily reflect use-cases where the cognitive load is induced by learning processes, which is an important aspect in the field of education. Further, some use-cases may be associated with stressful demands causing subjective strain, dysfunctional physiological activity, or deterioration of performance (Steeneken and Hansen, 1999), for example, due to the influence of time pressure. Although there exist potential speech parameters from the field of stress detection and emotion recognition based on the so-called Teager energy operator (e.g., Zhou et al., 2001; Nwe et al., 2003), they have not yet been used for automatic cognitive load recognition. Finally, automatic speech-based cognitive load recognition modelled so far was aimed at predicting discrete categories, usually comprising three load levels derived from the task difficulty. Interactive systems, in particular those employing multimodality, could benefit from more sensitive approaches providing deeper insights and more accurate user state determination. One way of achieving this is to treat cognitive load modelling as a regression problem, i.e., predicting a continuous numerical quantity instead of discrete categories.

1.3 Major Contributions

This thesis addresses the effectiveness of paralinguistic speech features for automatic cognitive load recognition. From the deficiencies discussed in the previous section, the following three aims have been defined for this thesis: (1) Development of a new speech-based cognitive load database, which reflects use-cases in which individuals are required to achieve learning goals; (2) Evaluation of traditional speech features and those from the field of stress detection for cognitive load recognition; (3) Development and evaluation of regression-based cognitive load analysis from speech.

Since these aims are accompanied by issues of optimisation, detailed elaborated concepts and development steps are required. The major contributions of this thesis can be summarised as follows:

- Development of a new database including speech recordings of individuals who performed a learning task and continuous numerical labels as the reference for cognitive load.
- Development of a software-framework for phoneme-based feature extraction and an activity detector for speech-event detection. Both tools perform prosodic analyses and extract features describing tempo, disfluency, and pausing in speech.
- Re-implementation and investigation of three Teager energy operator based features for speech-based cognitive load recognition. It is widely accepted that these features are better able to reflect the non-linear airflow structure of speech production under stressful conditions.
- Proposition of three new hand-crafted feature sets designed for automatic cognitive load recognition.
- The effectiveness of speech features for cognitive load assessment is demonstrated by a feature relevance analysis conducted using correlation as well as entropy-based measures.
- The effectiveness of speech features for cognitive load assessment is demonstrated by a systematic evaluation of cognitive load classification systems under task-dependent as well as task-independent conditions.
- The effectiveness of speech features for cognitive load assessment is demonstrated by the evaluation of regression-based cognitive load recognition systems.

Some of these contributions are reflected in the publications by the author of this thesis (Herms, 2016; Wirzberger et al., 2017, 2018; Herms et al., 2018).

1.4 Organisation of the Thesis

The remainder of the thesis is organised as follows. Chapter 2 introduces fundamental knowledge, which constitutes the basis for this thesis including the concept of cognitive load, the functioning of the human speech production system, and findings from the scientific literature regarding the effects of cognitive load on speech. Chapter 3 presents the cognitive load task designs and statistics of the speech material used for the investigations. In Chapter 4, methods for digital speech processing as well as a variety of indicative speech parameters are described and the process chain applied for feature extraction is presented. Then, in Chapter 5, three feature sets are proposed for cognitive load recognition and results of feature relevance analyses for different cognitive load tasks based on correlation as well as entropy-based measures are reported. In Chapter 6, the effectiveness of speech features is investigated by recognition experiments. This includes an extensive evaluation of cognitive load classification systems under task-dependent as well as task-independent conditions. Moreover, systems for regression-based cognitive load recognition are designed and evaluated. Finally, Chapter 7 summarises the contents of this thesis, highlights the contributions in conjunction with the main aims, and discusses future directions.

Chapter 2

Background

This chapter introduces necessary knowledge to familiarise the reader with the terms and concepts used throughout this thesis. First, this chapter starts with the basics of cognitive load (Section 2.1). Causal and assessment factors, working memory models, cognitive load theory principles, and different types of cognitive load are described. Next, cognitive load assessment approaches are reviewed in Section 2.2 including performance, subjective, physiological, and behavioural measures. Section 2.3 gives an introduction to the human speech production system and a literature survey of the effects of cognitive load on speech. In order to provide the reader with an impression of how the concept of cognitive load can be applied in practice, Section 2.4 presents various areas of application for cognitive load assessment, such as education, user interface design, car driving, and aviation.

2.1 Basics of Cognitive Load

The human cognitive system is considered to be an active, limited capacity information processing system, which is required for knowledge acquisition and understanding (Plass et al., 2010, p. 1). The limitations of the system refer to the human working memory (Section 2.1.2). This type of memory is necessary for holding and manipulating information while performing complex tasks.

The term ‘cognitive load’ refers to the amount of mental resources that are required for certain information processing tasks. It is a psychological construct that allows to explain and quantify human performance, learning, and cognitive processes. In formal terms, cognitive load is ‘the load imposed on an individual’s working memory by a particular (learning) task’ (Van Gog and Paas, 2012, p. 599). The characteristics of a task influence the performance and can induce additional effects. Highly demanding tasks can lead to cognitive overload, which can be considered a type of stress—that is, ‘a psycho-physiological state characterised by subjective strain, dysfunctional physiological activity, and deterioration of

performance' (Steeneken and Hansen, 1999). On the other hand, a less demanding task can lead to cognitive underload and, in turn, to boredom or a lack of motivation. It is often desirable to maintain an optimal level of cognitive load, because cognitive underload or overload may degrade the individual's performance; in the worst case, fatal errors occur in complex working environments (e.g., air traffic control centres). In the context of the cognitive load theory (Section 2.1.3), the level of cognitive load influences how much is learned and the complexity of what is learned (Paas et al., 2003b). Consequently, the goal, especially in instructional design, is to determine an appropriate level of cognitive load instead of its reduction (Brünken et al., 2010a, p. 255).

2.1.1 The Concept of Cognitive Load

In order to describe and assess cognitive load more precisely, it can be conceptualised by considering various aspects. Paas and Van Merriënboer (1994a) argued that the construct cognitive load consists of causal factors that affect cognitive load and assessment factors that are affected by cognitive load.

Figure 2.1 illustrates a schematic representation of the construct cognitive load including its factors. Causal factors refer to the characteristics of the subject (e.g., expertise level, age, and spatial ability), the characteristics of the task (e.g., complexity, time pressure, and pacing of instructions) in a given environment (e.g., noise and temperature), and the interaction between the subject and the task. The assessment factors comprise three different measurable dimensions: mental load, mental effort, and performance.

Mental load is related to the task and environmental demands. This dimension is assumed to be independent of the subject and constant for a given task in a given environment. Consequently, mental load can be regarded as an a priori estimate of cognitive load. *Mental effort* is related to the subject. This human-centred dimension reflects the amount of cognitive capacity that is actually allocated concerning the demands imposed by the task. The overall invested effort is assumed to be associated with all three causal factors, i.e., task characteristics, subject characteristics, and subject-task interactions. Theoretically, mental effort can be measured while the subject is involved in the problem solving or learning task (Paas et al., 2003b, p. 64). One can assume that mental effort includes important information that is not necessarily reflected in mental load or performance measures. *Performance*, the third measurable dimension, is specified by the achievements of the subject. Typical performance measures are the number of correct answers, number of errors, and time required. As with the mental effort, performance reflects all three causal factors and can be measured while the subject is working on the task.

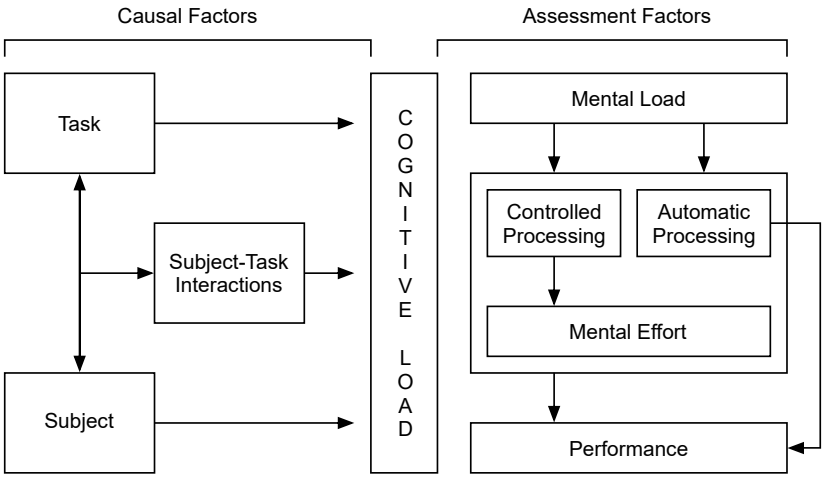


Figure 2.1 Schematic representation of the construct cognitive load including causal factors and assessment factors. Adapted from Paas and Van Merriënboer (1994a, p. 353)

2.1.2 Working Memory

The human cognitive architecture is assumed to include separate types of memory. Only the contents of the working memory can be monitored by humans in a conscious manner (Sweller et al., 1998, p. 252). This type of the human memory plays a crucial role for complex cognitive tasks such as reasoning, comprehension, and learning. The working memory provides temporary storage and manipulation of information (Baddeley, 1992, p. 1). Its limitations concerning capacity and time are well-known and widely accepted.

Miller (1956) introduced 'the magical number seven, plus or minus two', which was one of the first attempts to formalise the limits of the working memory capacity. The statement refers to the number of items that can only be processed at one time. In more recent works, it is reported that three to five items are processed simultaneously (cf. Cowan, 2001).

Different memory models were proposed over the years in order to explain the processes and functions of the human working memory system. In the following sections, some of the most influential models of working memory are discussed, namely the *multi-store model of memory*, *multi-component model of working memory*, and *embedded-processes model of working memory*.[1]

[1] Besides the three models presented in the following sections, there are a number of other models and approaches to describe how the working memory system works. For further information, the reader is referred to Miyake and Shah (1999).

2.1.2.1 Multi-Store Model of Memory

The argumentation for a distinction between a temporary short-term memory and a virtual long-term memory is based on empirical research (e.g., Brown, 1958; Peterson and Peterson, 1959). One of the first and most influential models, which considers the separation of human memory, was proposed by Atkinson and Shiffrin (1968). In this model, memory is divided into the three structural components: sensory register, short-term store, and long-term store.

Information from the environment enters the sensory register, where it resides for a very short period of time. The sensory information is initially processed and transferred to the short-term store. The transfer of information does not imply that the status in the original store is affected; rather, it can be understood as a process of copying (Shiffrin and Atkinson, 1969, p. 179). The short-term store controls the information flow into and out of the long-term store. It is assumed that information in the short-term store decays up to 30 seconds, but can be extended using a control process called rehearsal (Shiffrin and Atkinson, 1969, p. 180). In contrast to the sensory register and the short-term store, the long-term store can be regarded as a permanent repository for information.

The short-term store provides a number of useful functions (cf. Shiffrin and Atkinson, 1969). It decouples the memory system from the external environment and relieves the system from the responsibility of attention regarding environmental changes. Additionally, it provides a working memory for manipulations of information.

2.1.2.2 Multi-Component Model of Working Memory

Results of working memory investigations, also regarding patients with a defective short-term store, led to the conclusion that a single unitary component is not sufficient to characterise the architecture of a short-term store which acts as a working memory (Baddeley, 1992).

Instead, Baddeley and Hitch (1974) suggested that working memory represents a control system with its limits on storage and processing capabilities. Moreover, Baddeley (1992) argued that the working memory is divided into the three separate components: the central executive, phonological loop, and visuospatial sketchpad. The single components of this *multi-component model*, later complemented by the episodic buffer (cf. Baddeley, 2000), are assumed to work together as a working memory system.

The multi-component model is illustrated in Figure 2.2. Each component of the model functions as follows:

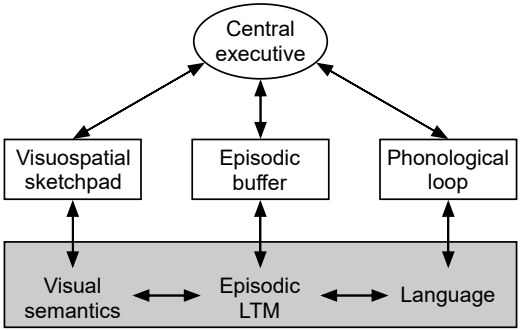


Figure 2.2 Baddeley's multi-component working memory model. The shaded areas refer to the long-term memory system. Adapted from Baddeley (2000, p. 421)

- The *central executive* controls and coordinates the slave systems (phonological loop and visuospatial sketchpad) (Baddeley, 1992). It is responsible for the attentional control. Thus, executive processes provided by this component are assumed to be important for the capacity to focus, to divide, and to switch attention (Baddeley, 1996, 2002).
- The *phonological loop* comprises two subcomponents, namely a temporary storage and subvocal rehearsal system. Acoustic or speech-based information in the phonological storage system decays after a few seconds unless it is revived through subvocal rehearsal. Furthermore, the subvocal rehearsal system has the function of registering visual information, such as words or pictures, within the phonological store (by subvocalisation) (Baddeley, 2003).
- The *visuospatial sketchpad* provides a temporary storage for the integration and manipulation of visual and spatial information (Baddeley, 2002). This component is assumed to be responsible for understanding complex systems (e.g., machinery), acquiring semantic knowledge about objects and how to use them, and spatial orientation as well as geographical knowledge (Baddeley, 2003).
- The *episodic buffer* is a limited-capacity temporary storage system, which integrates and combines visual and auditory information from the slave systems and from the long-term memory system (Baddeley, 2000). Thus, it serves as an interface between system components, each involving a different set of codes. The buffer can be accessed by the executive control, which binds the amount of information into episodes.

2.1.2.3 Embedded-Processes Model of Working Memory

An alternative approach to describe the working memory is provided by Cowan (1988) who formulated the *embedded-processes model*. This model defines working memory as an attentional system including processes that hold a limited amount of information in an accessible state in order to perform complex tasks. For this purpose, three components are hierarchically arranged (Cowan, 1999, p. 64): (1) the long-term memory that includes the amount of all stored information; (2) the activated memory (short-term store), which represents the currently activated part of the long-term memory; (3) the focus of attention (or awareness), which in turn is part of the activated memory. That is, the working memory is embedded in the long-term memory and includes activated data in the focus of attention. In addition, the model includes a brief sensory store, which stores sensory stimuli for only several hundred milliseconds.

It is assumed that attention is limited in capacity (about three to five items), while information activation is limited in time (about two to thirty seconds) unless it is reactivated through additional, related stimulus presentations or thought processes (Cowan, 2001, p. 92). The focus of attention is controlled using voluntary processes caused by the executive control as well as involuntary processes, for example, induced by novelty or stimuli with significant signals. Recurring and unchanged context information still activates some processes in memory but may refer to habituation, which facilitates the control of awareness for new relevant information (Cowan, 1999, p. 67). Depending on the situation, different data are prioritised for different processes of the working memory system. Complex tasks may require associations of particular information from all three sources (long-term memory, activated memory, and focus of attention) that can be brought and kept into the focus of attention.

2.1.3 Cognitive Load Theory

Cognitive load theory (CLT) is concerned with the development of instructional design principles and strategies. The goal of instruction is to increase the amount of knowledge in long-term memory by learning (Sweller et al., 2011, p. 24). The theory suggests that appropriate learning can be realised by the consideration of the human cognitive architecture, i.e., it assumes a limited working memory capacity and a virtual unlimited long-term memory (Paas et al., 2004). The focus of CLT is on the control of cognitive load in order to reach useful learning outcomes in complex cognitive domains. In other words: it is assumed that learning can be supported by imposing adequate levels of cognitive load based on measurements that refer to the limitations of the working memory capacity (Kirschner, 2002). Within the framework of CLT, there are no explicit assumptions about the

architecture of working memory and its relation to cognitive load (Brünken et al., 2010a, p. 261). Nonetheless, Baddeley's multi-component model seems to be the best-known working memory model and, in addition, is also referenced by the cognitive load theory literature (cf. Sweller et al., 2011).

2.1.3.1 The Process of Learning

Learning can be defined as 'a positive change in long-term memory' (Sweller et al., 2011, p. 53). It is an active and resource consuming process in working memory resulting in the construction of schemas that are stored in long-term memory (Brünken et al., 2010a, p. 254). This process involves the acquisition of information and the comprehension of how multiple information elements interact (Sweller et al., 2011, p. 65). In this context, the term 'element' refers to anything that is learned or processed. Interacting elements are defined as 'elements that must be processed simultaneously in working memory because they are logically related' (Sweller et al., 2011, p. 58).

The concept of 'schema' arose in psychology through the works of Piaget (1923) and Bartlett (1932). A schema can be regarded as a cognitive construct that combines multiple information elements to a single element, which has a specific function (Paas et al., 2003a). It can be transferred from long-term memory to the working memory and treated as only one element consisting of a number of interacting elements. Schemas can be formed during problem solving by creating chunks (combining information elements), by complementing information elements in already existing schemas, or by obtaining schematised information from other people (Van Merriënboer and Sweller, 2010).

Constructed schemas may become automated if they are frequently used in practice. For instance, learning how to read or write requires combining single letters into words, combining words into phrases and so forth. Schema automation can be helpful for aspects which are consistent across different tasks and, in turn, can free the capacity of the working memory for other activities. Thus, besides the function of learning to store information in the long-term memory, schema acquisition and automation have the effect of reducing the working memory load (Sweller et al., 2011, 58).

In order to achieve learning goals, the following five psychological processes have to be considered and work together (Clark et al., 2011, pp. 36–37):

1. *Attention*: The Learner must focus on information that is relevant to the learning goal.
2. *Activation of prior knowledge*: Relevant pre-existing schemas are transferred from long-term memory into working memory.

3. *Elaboration-rehearsal*: The working memory processes new knowledge and skills in order to integrate them into activated schemas.
4. *Encoding*: The content is transformed resulting in expanded schemas stored in long-term memory.
5. *Retrieval*: The new schemas are brought back into working memory when needed.

Various cognitive load effects, that show specific instructional implications, have been revealed over the years. These effects are intended to indicate how to provide information that best facilitates learning. Some examples are *split-attention*, *modality*, *redundancy*, and *expertise reversal*. For a deeper discussion, the reader is referred to Sweller et al. (2011).

2.1.3.2 Types of Cognitive Load

Managing the working memory load requires the differentiation of its possible causes. The development of CLT has undergone three main stages in which specific types of cognitive load had been introduced (Moreno and Park, 2010): (1) The first stage focused on 'other mental activities' that must remain limited to avoid cognitive load which interferes with learning. This source of load is referred to as *extraneous* cognitive load; it can be reduced by better instructional design. (2) In the second stage, an additional source of load was considered along with extraneous load, the so-called *intrinsic* load, which is imposed by the basic characteristics of information and cannot be reduced by instructional design. (3) Finally, the third stage introduced the third source of load, namely *germane* cognitive load, which has a positive relationship with learning and can be increased by instructional design.

In summary, the gradual evolution of CLT results in the distinction between three types of cognitive load. The relationship between the working memory capacity and the different types of cognitive load is illustrated in Figure 2.3. A closer look at each individual type is given below:

- *Intrinsic load* origins from cognitive processing related to the complexity of information in a task. The degree of this type of cognitive load depends on the number of cognitive elements that are simultaneously processed in working memory, but also on the prior knowledge of the learner (Schnotz and Kürschner, 2007, p. 476). For instance, learning the syntax of a new language is strongly linked with analysing how words are related to each other. Consequently, the element interactivity is rather high. In contrast, learning a list of vocabulary leads to a low element interactivity, because elements are not processed simultaneously in working memory. Considering the learner's prior knowledge in a specific task, intrinsic load is fixed regarding the material being dealt with (Paas and Sweller, 2014, p. 37).

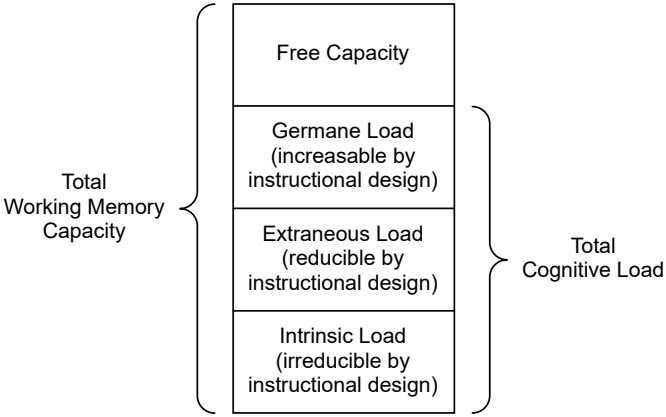


Figure 2.3 Working memory capacity and different types of cognitive load. Adapted from Moreno and Park (2010, p. 18)

- *Extraneous load* refers to the unnecessary cognitive load caused by the design of the instruction and learning material (Sweller, 2010b, p. 42). In effect, this type of load does not contribute to knowledge acquisition. Schnotz and Kürschner (2007, p. 476) stated that extraneous load is concerned with an unnecessarily high degree of element interactivity in working memory, for example, caused by redundant information such as additional text which is integrated into a self-explanatory diagram. On the other hand, extraneous load can be the result from irrelevant cognitive activities which are not aimed at schema acquisition and automation. Both assumptions have one thing in common: extraneous load interferes with learning and should be reduced as far as possible.
- *Germane load* is related to the effort which contributes to schema construction (Sweller et al., 1998, p. 259). This type of cognitive load is relevant and should be increased as far as possible because it facilitates learning. In this connection, an instructional design which results in unused working memory capacity, for example, caused by low intrinsic or low extraneous load, can be further improved by encouraging learners to focus on schema construction (Sweller et al., 1998, p. 264).

The theory assumes that intrinsic, extraneous, and germane cognitive load are additive, i.e., the sum of all three types constitutes the total cognitive load (Paas et al., 2003b, p. 65). The intrinsic load is fixed regarding the material and the learner's prior knowledge, but it can be reduced through schema acquisition and automation of the learner (Paas et al., 2003b). The remaining working memory

capacity can be allocated by extraneous and germane load in a balanced way. The reduction of extraneous load may increase the germane load caused by effective instructional design. Adding a heavy extraneous load to a heavy intrinsic load may result in interference with learning because there is no capacity available for germane load. It should be noted that the effects of cognitive load due to extraneous load can be demonstrated only in scenarios with a high number of elements that must be simultaneously processed in working memory (Paas and Sweller, 2014, pp. 38–39).

Due to the lack of facet-specific empirical results for the separation of cognitive load types, there is an ongoing discussion in this regard. For instance, it has been suggested that germane load is not an independent type of cognitive load but can be considered as related to specific learning goals and associated with intrinsic cognitive load (Sweller, 2010a; Kalyuga, 2011). It has also been argued that the total load experienced cannot simply be regarded as the sum of the three different types. There might be different possible interpretations in terms of the relationship between intrinsic and germane load: if intrinsic and germane load are seen as members of different ontological categories, i.e., 'material' and 'cognitive processes', respectively, there are principled objections to adding the two together (De Jong, 2010).

2.2 Cognitive Load Assessment

Cognitive load can be regarded as a theoretical construct describing information processing of the human cognitive system which is not directly observable. Nevertheless, in order to deploy cognitive load assessment in laboratory environments or real-life use-cases, there is a need for valid and reliable measuring instruments.

As pointed out in Section 2.1.1, the assessment factors of cognitive load include three different dimensions, namely mental load, mental effort, and performance. Basically, measurement approaches can be classified into analytical and empirical methods (Xie and Salvendy, 2000). Analytical methods are rather concerned with the mental load. These types of methods are used in a predictive as well as an evaluative way and contain techniques such as comparison, consulting expert opinion, utilisation of mathematical models, task analysis, and computer simulation. In contrast, empirical methods are concerned with post-hoc measures of mental effort and performance. These methods contain operator opinion, primary task, secondary task, and physiological techniques.

Another approach to classifying methods of assessing cognitive load has been proposed by Brunken et al. (2003). Table 2.1 shows this classification of methods including the two dimensions: objectivity (subjective or objective) and causal relationship (direct or indirect). The objectivity dimension distinguishes

Table 2.1 Classification of methods for the assessment of cognitive load based on objectivity and causal relationship. Adapted from Brunken et al. (2003)

	Causal relationship	
Objectivity	Indirect	Direct
Subjective	Self-reported invested effort	Self-reported stress level Self-reported difficulty of materials
Objective	Physiological measures Behavioural measures Learning outcome measures	Brain activity measures Dual-task performance

between methods that use subjective data, which are self-reported, and objective observations of behaviour, physiological conditions, or performance. Regarding the causal relationship, one can distinguish between methods in which the parameter of interest is related to cognitive load in either a direct or indirect manner. For instance, a direct relationship exists between the difficulty of the task and cognitive load, because this difficulty is caused by intrinsic and extraneous load. On the other hand, learning outcome measures are indirect, because they depend on information processing that may be affected by cognitive load.

According to the type of measuring instruments, techniques can be classified in terms of performance, subjectivity, physiology, and behaviour (Sweller et al., 1998; Brunken et al., 2003). The following sections describe these aspects in more detail.

2.2.1 Performance Measures

The assessment of cognitive load using performance measures is based on the assumption that there is a relationship between the task performance and the level of cognitive load experienced by the subject. Cognitive load, which is involved in a particular task, can be evaluated using a variety of performance metrics. Typical parameters are the number of correct answers, number of errors, and time required. Nevertheless, such parameters are not always reliable in a single-task condition and, consequently, the results should be interpreted very carefully. More precisely, the relation between the load imposed by the task and the performance of the subject depends on task demands, human strategies, motivation, and individual characteristics (Paas et al., 2005; Cegarra and Hoc, 2006).

An alternative approach is to introduce a second task, which is conducted concurrently with the main task. The underlying idea of this dual-task paradigm

is that the available capacity, which is not allocated in connection with the primary task, can be employed for a second task. The secondary task can be considered to be a cognitive distraction. It typically includes simple activities regarding the attention of the subject, such as detecting visual or auditory signals or remembering a series of numbers (Sweller et al., 2011, p. 78). The dual-task approach can be applied in two different ways (Brunken et al., 2003): (1) the secondary task is added to the primary task with the goal to induce working memory load, whereby the performance related to the primary task is of interest; (2) the secondary task is used to measure the working memory load induced by the primary task, i.e, the variable of interest is the performance of the secondary task.

A more difficult primary task can result in an increase in allocated working memory resources, meaning that less capacity is available for the second task. Consequently, the performance of the second task decreases and can be used as an indicator for cognitive load assessment. The main advantage of applying the dual-task method is that it can provide an almost continuous measure while the subject is working on the task (Sweller et al., 2011, p. 80). Moreover, using the same conditions for a secondary task, different primary tasks can be compared regarding the task design and cognitive load. However, there are limitations concerning the dual-task approach. Although it forms a sensitive and reliable technique, the secondary task can interfere with the primary task, especially if the primary task is very complex or in cases where working memory capacity is particularly limited, such as in the elderly (Van Gerven et al., 2006).

2.2.2 Subjective Measures

Subjective measurement in conjunction with self-report rating scales is a common approach for the assessment of cognitive load. The scale method is based on the assumption that users are able to introspect on their cognitive processes and to report their perception of invested mental effort (Brünken et al., 2010b, p. 182). For instance, subjects are asked to rate their mental effort on a 9-point Likert scale ranging from ‘very, very low mental effort’ to ‘very, very high mental effort’ (Paas et al., 1994). Such a unidimensional scale assesses the overall cognitive load without considering additional factors.

Variables that are indirectly related to cognitive load such as frustration or fatigue can also be important indicators (Paas et al., 2003b). The involvement of many variables forms a multidimensional assessment tool, which serves as a reliable measuring instrument. For instance, the *NASA Task Load Index* (Hart and Staveland, 1988) is used to derive a sensitive estimate of workload by combining the following six rating scales: performance, mental effort, frustration level, task demand, physical demand, and temporal demand.

According to CLT, some attempts have been done to assess various aspects of cognitive load imposed by a learning task. Cierniak et al. (2009) introduced a six-point subjective rating scale to measure the three types of cognitive load—intrinsic, extraneous, and germane load—in order to explain learning outcomes. In this connection, the intrinsic load scale asked 'How difficult was the learning content for you?', the extraneous load scale asked 'How difficult was it for you to learn with the material?', and the question in terms of germane load was 'How much did you concentrate during learning?'. Moreover, Leppink et al. (2013) developed an instrument for the measurement of the three cognitive load types in complex knowledge domains such as statistics. They introduced a questionnaire with a rating scale ranging from '0' ('not at all the case') to '10' ('completely the case'). The first three questions are related to intrinsic load (e.g., 'The activity covered formulas that I perceived as very complex'), the next three to extraneous load (e.g., 'The instructions and/or explanations were full of unclear language'), and the last four questions are related to germane load (e.g., 'The activity really enhanced my understanding of the formulas covered'). More recently, Klepsch et al. (2017) proposed two types of questionnaires: informed rating and naïve rating. In the former case, participants are first qualified to understand and differentiate the three types of cognitive load; after conducting each task, learners have to rate all three types of load on a 7-point Likert scale from 'very low' to 'very high'. Regarding the naïve rating questionnaire, participants are required to rate all three types of load, each represented by questions, without being informed about the concept of cognitive load—the naïve rating questionnaire is also provided with a 7-point Likert scale (from 'absolutely wrong' to 'absolutely right') and comprises two questions related to intrinsic load (e.g., 'This task was very complex.'), three related to extraneous load (e.g., 'The design of this task was very inconvenient for learning.'), and another three items related to germane load (e.g., 'I made an effort, not only to understand several details, but to understand the overall context.').

Krell (2015) developed a testing instrument to measure the cognitive load factors mental load and mental effort separately. This instrument consists of six items for each factor, while each item is provided with a rating scale ranging from '1' ('not at all') to '7' ('totally'). The items for mental load ask to indicate the complexity of the task (e.g., 'The tasks were challenging'), whereas items for mental effort focus on the personal effort (e.g., 'At the reply to the tasks I have made an effort intellectually').

The advantage of subjective measures is attributed to its simple utilisation. However, the assessment of cognitive load based on the rating scale method is usually conducted post-hoc. Subjects are asked to rate their experienced cognitive load after the task has finished. The result is a global scaling across different parts of the working procedure. Thus, it remains unclear how the actual cognitive load

varies over time and which specific aspects caused the level of cognitive load (Brünken et al., 2010b). Nevertheless, subjective measures can also be applied in a synchronised way during the working procedure, for example, by using a pop-up window in a multimedia scenario (cf. Tabbers et al., 2004).

2.2.3 Physiological Measures

Active research has revealed that variations in human cognitive processing are reflected in physiological measures (Kramer, 1991). These types of measurement techniques provide insights into the changes in bodily functions. Physiological signals can be acquired continuously allowing the cognitive load to be assessed at a high sampling rate.

There are numerous physiological measuring methods that may give an objective reference for cognitive load. Typical measures that have been investigated in connection with cognitive load include brain activity monitored using electroencephalography (EEG) (e.g., Wilson and Russell, 2003) or functional near infrared (e.g., Izzetoglu et al., 2003), heart rate and heart rate variability (e.g., Mulder, 1992; Kennedy and Scholey, 2000; Nickel and Nachreiner, 2000), galvanic skin response (or conductance response) (e.g., Jacobs et al., 1994; Shi et al., 2007), and eye activity such as the pupillary response, blink rate, and eye fixation (e.g., Backs and Walrath, 1992; Lipp and Neumann, 2004; Ryu and Myung, 2005; Reyes and Lee, 2008; Siegle et al., 2008).

Some relationships between physiological measures and cognitive load could be found in the past. The work of Antonenko and Niederhauser (2010) showed better learning outcomes when adding leads to hypertexts (hypertext node previews). This could be affirmed only by EEG measures (alpha, beta, and theta brainwave rhythms) that are sensitive enough to show differences in the instantaneous cognitive load. In contrast, the self-report mental effort measure was not sensitive enough to show effects due to its association with the overall cognitive load. It is well-known that heart rate increases and heart rate variability decreases during effortful mental processing (Mulder, 1992). Paas and Van Merriënboer (1994b) demonstrated in their work that spectral information on heart-rate variability is not sensitive to the differences in cognitive load. The authors Shi et al. (2007) analysed the mean galvanic skin response across subjects; results confirm that there is an increase as cognitive load increases. Van Gerven et al. (2004) showed that pupil dilation increased according to the increased level of cognitive load, but there was no correlation for elderly participants.

Although physiological techniques seem to provide promising measures for the assessment of cognitive load, there are some limitations. In order to obtain valid information, some criteria for the selection of a technique should be considered, for example, sensitivity, intrusiveness, reliability, and generality of

application (Kramer, 1991). Furthermore, artefacts or continuous noise can be included in the physiological signals, which makes data analysis more complicated with respect to the estimation of cognitive load.

2.2.4 Behavioural Measures

Behavioural measurements can be obtained from any user activity related to the task. The methods used are based on the assumption that behavioural patterns differ under different cognitive load levels. The behavioural data is objective and can be collected through non-intrusive monitoring, for example, using cameras or microphones that are integrated into a display device. Consequently, users do not notice that they are being observed—they can devote themselves fully to the task.

For instance, Gütl et al. (2005) proposed an adaptive e-learning framework, which considers users' learning activities by monitoring eye movements (e.g., fixation and gaze duration) in real-time. Other works proposed behavioural measures based on input devices, such as the mouse and keyboard, to derive the users' emotional state and to provide adaptive systems (e.g., Ark et al., 1999; Liu et al., 2003). Oviatt (2006) explored the usability of different interfaces in connection with digital-pen based gesturing. In this connection, different user-centred design principles and strategies and the manner how these enhance users' performance had been illustrated.

The amount of time needed to solve a particular task is also assumed to be an important indicator since all cognitive processes take time (Brünken et al., 2010b, p. 188). For instance, Dubé and McEwen (2015) used the response latency to investigate behavioural aspects of touchscreen interactions and their consequences for the conceptual understanding of content. Moreover, Pouw et al. (2016) analysed the reaction time along with accuracy to estimate the subjects' performance in assessing mechanical concepts.

Apart from that, information retrieval patterns can be used as a measure of cognitive load; the way in which learners search and select information can serve as an indicator (Möller and Müller-Kalthoff, 2000). For instance, novice learners are unfamiliar with the domain and might start reviewing basic information contained in hypertext pages, whereas domain experts might directly navigate to pages that include specific information. The analysis of navigation pathways in such environments could then reveal schema construction.

Most of the discussed techniques seem to be reliable, but they raise the question of validity. There could be several causes for changes in the parameters, such as information complexity, interest, or readability (Brünken et al., 2010b, p. 187). Thus, behavioural data analysis requires careful planning to ensure the validity of cognitive load measures.

Fundamentally, measures based on linguistic and paralinguistic speech parameters can also be classified as behavioural measures, because they uncover the manner how something is spoken regardless of the meaning of the utterance (Chen et al., 2016, p. 22). For further information on cognitive load in connection with speech parameters, see Section 2.3.

2.3 Cognitive Load and Speech

Speech is a complex signal containing a multitude of information that may contribute as indicators for cognitive load assessment. The parameters in the speech signal can be measured in a contact-free, non-intrusive way; the recording process can be carried out without the monitored person perceiving it. Hence, the speech signal is a promising source for cognitive load monitoring.

Before taking a closer look at the literature regarding the effects of cognitive load on speech, the fundamentals of the human speech production system are introduced.

2.3.1 The Human Speech Production System

Speech can be seen as a natural way of communication for human beings. Although speech is used in our daily life, it is a very complex process comprising many steps from cognitive activities to signal production. Honda (2003) describes the speech production mechanisms with the following four processes:

1. Transformation of utterances into phonemes in the language centre of the brain,
2. Generation of vocal organ related commands in the motor centre of the brain,
3. Movement of the vocal organs based on the generated motor commands, and
4. Air emission caused by the lungs in order to produce speech.

2.3.1.1 The Human Vocal Apparatus

The human vocal apparatus is shown in Figure 2.4. It consists of various organs that are excited in order to characterise the sound being produced. First, air is absorbed by human beings as part of the breathing procedure. The air pressure gets released by muscle force from the lungs through the trachea and passes the articulatory organs. The larynx, commonly known as Adam's apple, contains two small folds of muscle, the so-called vocal folds. These two folds can be moved together or apart; the space between them is defined as the glottis (Jurafsky and Martin, 2009, p. 252). If the vocal folds are tensed, the air pressure causes their vibration and the air flow is chopped into quasi-periodic pulses by the vocal

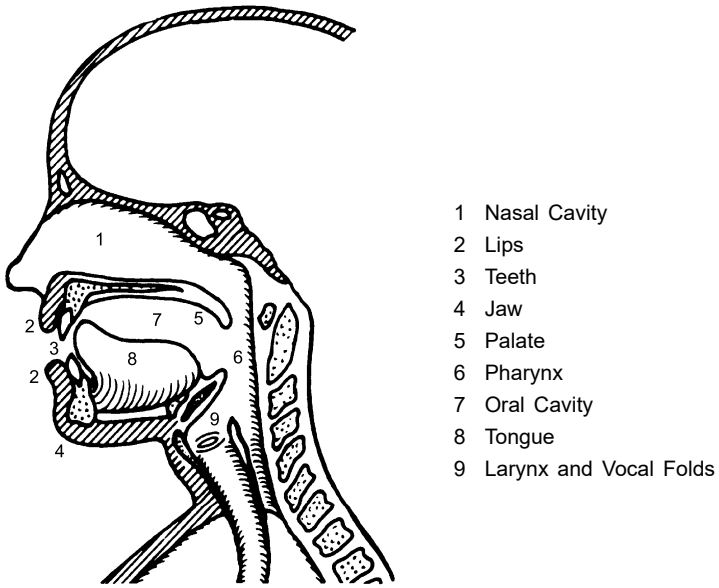


Figure 2.4 The human vocal apparatus. Adapted from Pfister and Kaufmann (2008, p. 13)

folds. This vibration leads to the production of voiced sound units with a specific frequency, which is also known as the fundamental frequency. Higher frequencies are caused by thinner and longer vocal folds, whereas in low frequencies, the vocal folds become shorter and thicker. The frequency ranges from about 80 Hz to 400 Hz for male adult speakers and about 120 Hz to 800 Hz for female adult speakers (Honda, 2008, p. 12). Unvoiced sounds are produced when the vocal folds are relaxed (i.e., no vibration) and the air flow passes through a constriction in the vocal tract.

The produced acoustic signals from the vocal folds are changed in terms of the sound by the vocal tract (pharynx, oral cavity, and nasal cavity). Additionally, the sound caused by air turbulence is also formed by the vocal tract. In this connection, the vocal tract acts as an acoustic filter. It is to note that the movements of the articulators (tongue, lips, jaw, and palate) changes the transfer function and, thus, the resonant frequencies of this acoustic filter (Pfister and Kaufmann, 2008, pp. 12–13). Consequently, a generated sound at a particular point in time depends on the state of the vocal folds as well as the position, shape, and size of the different speech organs. Finally, speech is composed of a sequence of these generated sounds.

2.3.1.2 The Source-Filter Model

From the perspective of acoustic theory, it is common to describe the human speech production using Fant's source-filter model (Fant, 1960). This model, illustrated in Figure 2.5, explains the acoustics of a speech sound including how pulses are produced by the glottis (the source) and afterwards shaped by the vocal tract (the filter). The source-filter model unifies a series of linear time-invariant systems in order generate a speech signal.

Signal modifications can be described as a function based on the concept of convolution, a mathematical operation, which expresses the input-output relationship of a system. Let $x(n)$ and $h(n)$ be two sequences that are related to the values of $y(n)$, one can say that $y(n)$ is the convolution of $x(n)$ with $h(n)$ and represent this by the notation $y(n) = x(n) * h(n)$. The result $y(n)$ is a sequence, where each value is commonly called the convolution sum (Oppenheim et al., 1999, p. 23):

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k). \tag{2.1}$$

The speech signal can be formulated as a convolution by $f(n) = u(n) * v(n) * r(n)$. The corresponding z-transform of the signal, with z being the complex variable, is defined as

$$F(z) = U(z) \cdot V(z) \cdot R(z), \tag{2.2}$$

where the function $U(z)$ refers to the source, which can be either voiced or unvoiced, $V(z)$ refers to the vocal tract model, and $R(z)$ refers to the radiation model representing the pressure at the lips (Schukat-Talamazzini, 1995, p. 33).[2]

In essence, speech sounds can be classified as either voiced or unvoiced. Voiced sounds are modelled by signals that are generated with the period T_g and transferred to the glottal pulse model $G(z)$. The result is an impulse response $g(n)$ with the glottal wave shape. In order to describe the production of a voiced speech signal instead of a speech signal in general, the glottal pulse, vocal tract, and radiation components are combined all together and represented as a single transfer function (Rabiner and Schafer, 1978, p. 104):

$$H(z) = G(z) \cdot V(z) \cdot R(z). \tag{2.3}$$

The intensity of voiced sounds is adjusted by the amplitude control parameter A_v. Unvoiced sounds are approximated by flat-spectrum noise in connection with the amplitude control parameter A_{uv} (Rabiner and Schafer, 1978, p. 103).

[2]For details on linear time-invariant systems, discrete-time convolution, and z-transform of signals, see Oppenheim et al. (1999).

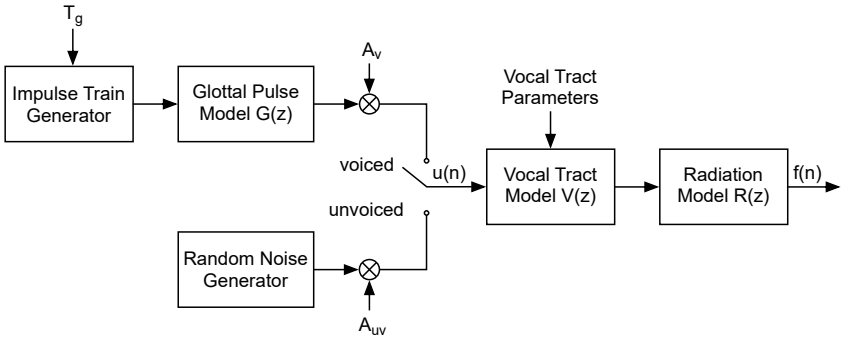


Figure 2.5 The source-filter model for human speech production. Adapted from Rabiner and Schafer (1978, p. 105)

For a deeper discussion on acoustic theory and speech production modelling, the reader is referred to Rabiner and Schafer (1978).

2.3.1.3 Phonemes

Speech is individual, i.e., the same spoken unit articulated by different speakers results in differences of the speech signal. Even if the same speaker tries to pronounce the same unit several times, there are differences in the signal. Single spoken units can be described by the terms 'phone', 'allophone', and 'phoneme'. A phone is the smallest unit of speech sounds. It is characterised by the timbre, duration, accentuation, and pitch (Fellbaum, 2012, p. 67). Some phones can have the same meaning. Allophones are different phones, which are assigned to the same phoneme. A phoneme is a symbolic unit of speech at a particular level of representation (Harper and Maxwell, 2008, p. 801). These phonetic symbols can be used to transcribe the sounds of a spoken language and provide, in a combined form, a description of how words are pronounced.

In order to apply a set of phonemes for diverse areas of application, several phonetic alphabets have been introduced in the past. The International Phonetic Alphabet (IPA, 1999) was suggested as a standard by the *International Phonetic Association* and is commonly used by linguists. This alphabet has been established in order to provide an accurate system for transcribing speech sounds of all human languages. It is based primarily on the Latin alphabet. The ARPAbet (Jurafsky and Martin, 2009, pp. 250–282) is another phonetic alphabet, which was developed by the *Advanced Research Projects Agency* (ARPA) and specifically designed for American English. This alphabet is often used in applications such as automatic speech recognition systems because it is based on ASCII—a

Table 2.2 IPA, ARPAbet, and SAMPA symbols for the transcription of English consonants and vowels. Adapted from Gibbon et al. (1997, pp. 688–689), Wells (2000), and Jurafsky and Martin (2009, pp. 251–252)

Consonants				Vowels			
IPA	ARPAbet	SAMPA	Word	IPA	ARPAbet	SAMPA	Word
p	p	p	parsley	i	iy	i	lily
t	t	t	tea	ɪ	ih	I	lily
k	k	k	cook	eɪ	ey	eI	daisy
b	b	b	bay	ɛ	eh	E	pen
d	d	d	dill	æ	ae	{	aster
g	g	g	garlic	ɑ	aa	A	poppy
m	m	m	mint	ɔ	ao	O	orchid
n	n	n	nutmeg	ʊ	uh	U	wood
ŋ	ng	N	baking	oʊ	ow	o	lotus
f	f	f	flour	u	uw	u	tulip
v	v	v	clove	ʌ	ah	V	cup
θ	th	T	thick	ɝ	er	3`	bird
ð	dh	D	those	aɪ	ay	aI	iris
s	s	s	soup	aʊ	aw	aU	flower
z	z	z	eggs	oɪ	oy	OI	soil
ʃ	sh	S	squash	ə	ax	@	lotus
ʒ	zh	Z	ambrosia				
tʃ	ch	tS	cherry				
dʒ	jh	dZ	jar				
l	l	l	licorice				
w	w	w	kiwi				
r	r	r	rice				
j	y	j	yellow				
h	h	h	honey				

Note that some rarer symbols were omitted

suitable format for computational representations of pronunciations. The *Speech Assessment Methods Phonetic Alphabet* (SAMPA) (Wells, 1997) serves as another example. It has been developed within the European ESPRIT project SAM. As with the ARPAbet, SAMPA uses ASCII symbols, but it is primarily intended for the European language area (Fellbaum, 2012, p. 69).

Table 2.2 gives an overview of the phonetic alphabets IPA, SAMPA, and ARPAbet and, furthermore, distinguishes between consonants and vowels. Consonants are sounds that are produced in conjunction with a partial or complete closure of the vocal tract; as a consequence, respiratory air is hindered or stopped. Depending on the type of consonants, the vocal folds vibrate or rest. Vowels are

sounds that are articulated with vibrating vocal folds while the respiratory air passes unhindered through the mouth or both the mouth and the nose (Pfister and Kaufmann, 2008, pp. 14–15).

Languages can be characterised by syllables, which are larger units than a single sound. However, there is no general definition for syllables (Jurafsky and Martin, 2009, p. 257). A syllable can be described as a vowel-like sound together with some surrounding consonants that are associated with it. The process of syllabification, i.e., breaking up a word into syllables, involves the separation of a sequence of consonants and vowels into substructures. For a deeper insight into syllabification, the reader is referred to Jurafsky and Martin (2009).

2.3.2 Effects of Cognitive Load on Speech

Previous studies in literature carried out statistical analyses in order to reveal the influence of cognitive load on speech-based parameters (i.e., 'features'). In this connection, some studies refer to the concept of stress or cognitive stress. Jameson et al. (2010) argued that stress is linked to cognitive load and time pressure because these resource limitations can be causes as well as consequences of stress. Scherer et al. (2002) assumed that in contrast to a single task condition, dual-task scenarios including attentional demands can induce psychological stress due to the experienced cognitive capacity limitations under time pressure. Time pressure can lead to emotional reactions, which, in turn, can have an indirect effect on cognitive load (Galy et al., 2012). Consequently, the review of the literature, presented in the following sections, will also include studies concerning stress due to cognitive tasks.

2.3.2.1 Linguistic Features

The exploration of linguistic features is motivated by the hypothesis that different levels of cognitive load result in different linguistic patterns. Since this thesis does not focus on linguistic features, only a short summary is given.

Khawaja et al. (2012) reported results regarding linguistic and grammatical features in the context of collaborative communication in bushfire management teams. They showed that high task load leads to an increase in word count, words per sentence, disagreement words, and first as well as third-person plural pronouns. Furthermore, a decrease in agreement words and first as well as third-person singular pronouns could be observed.

Similar results are reported in the work of Sexton and Helmreich (2000). They demonstrated that the number of words increased during periods of high workload in a flight crew communication scenario. In this connection, the captain

Table 2.3 Review of the effects of cognitive load on linguistic features

Feature	Tendency
Word count	+
Words per sentence	+
Agreement words	−
Disagreement words	+
First-person singular pronouns	−
First-person plural pronouns	+
Third-person singular pronouns	−
Third-person plural pronouns	+

+: feature value increases as cognitive load increases,
−: feature value decreases as cognitive load increases

of the crew used more first-person plural for team building purposes, especially in stressful situations (e.g., 'we need to ...,' or 'our problem ...,').

Table 2.3 gives a summary of the effects of cognitive load on linguistic and grammatical features. A comprehensive investigation including pauses, linguistic features, grammatical features, and language complexity in conjunction with cognitive load can be found in Khawaja (2010).

2.3.2.2 High-Level Speech Features

Previous studies investigated the influence of cognitive load on high-level speech features. Some of these parameters can be classified as prosodic features due to the characteristics in terms of disfluency and tempo in speech.

For instance, Müller et al. (2001) analysed the parameters disfluency, articulation rate, content quality, and the number of syllables, silent pauses as well as filled pauses. Berthold and Jameson (1999) examined, in addition to these features, the properties of the symptom sentence fragments. The effects of high cognitive load on the duration of voiced segments in speech are shown in Boril et al. (2010). Potential concerning the three speech features pause length, pause frequency, and response latency are reported in the work of Khawaja et al. (2007). Moreover, Khawaja et al. (2008) investigated pausing in speech by computing the percentage of silent pauses, filled pauses, and total pauses.

Berthold and Jameson (1999) reviewed the literature of these aspects in speech as well as the corresponding tendencies under cognitive load. The results are shown in Table 2.4. It can be seen that cognitive load causes a reduction in output quality, which is reflected by errors, such as the number of sentence fragments, false starts, and repetitions. Sentence fragments can be defined as an incomplete syntactic structure, i.e., the speaker gives up talking or begins a new sentence

Table 2.4 Review of the effects of cognitive load on high-level speech features. Adapted from Berthold and Jameson (1999, p. 237)

Feature	Tendency
Sentence fragments (number)	+
False starts (number)	+
Repetitions (number)	+
Articulation rate	−
Speaking rate	−
Onset latency (duration)	+
Silent pauses (number)	+
Silent pauses (duration)	+
Filled pauses (number)	+
Filled pauses (duration)	+

+: feature value increases as cognitive load increases,
−: feature value decreases as cognitive load increases

during the articulation. False starts are a specific type of self-repair, which begins with an alternative formulation of the content. A typical effect of cognitive load on speech is a decrease in the speaking rate (number of spoken units per time) as well as the articulation rate (number of spoken units per time excluding pauses). The above-mentioned studies show the same effects compared to the literature review of Berthold and Jameson (1999). General assumptions should, however, be made carefully. There are also studies showing other effects regarding the tempo in speech. For instance, Lively et al. (1993) demonstrated that utterances produced under workload were shorter in duration. Similar results are reported in Scherer et al. (2002) showing a shorter average duration of syllables (higher speaking rate) under high load as well as high-stress conditions.

Pauses in speech have the potential to be indicative of cognitive load (cf. Müller et al., 2001; Khawaja et al., 2007, 2008). In essence, there are three different types of pauses in speech: onset latency, silent pauses, and filled pauses. Onset latency is the time length between the stimulus and the onset of the first spoken unit (e.g., phoneme or syllable). In contrast, silent pauses and filled pauses (e.g., 'uh' or 'uhm') result in disfluencies during the articulation. Esposito et al. (2007) stated that pauses occur during speech flow in the form of a planning process which is not performable during the articulation phase. Moreover, the number and duration of pauses are reflected by the mental effort related to the lexical and semantical complexity. Besides the cognitive aspects, pauses can be linked to physiological (breathing activity), socio-psychological (speaking anxiety), or communicative (rhetorical or provide the opportunity to comprehend the message) causes (cf. Schilperoord, 2002, p. 75). Schilperoord (2002) argued

that the duration of pauses may be regarded as the response time, which, in turn, can be considered to be the cognitive processing time. This assumption can lead to the conclusion that long pauses are attributed to higher levels of cognitive load. However, a direct transformation of pause time to cognitive processing time is not always useful. Pauses are directly observable and can be measured in a precise manner which is not the case for cognitive load. For instance, pauses of two seconds would simply result in a load, which is twice as heavy as the load that is reflected by pauses of one second. Nevertheless, one can assume that more and longer pauses in speech reflect more mental effort.

2.3.2.3 Low-Level Speech Features

Besides non-acoustic speech parameters, previous works also investigated the effects of cognitive load on low-level signal characteristics. A summary is given in Table 2.5.

Cognitive load may induce stress and can result in physiological consequences, such as irregular breathing and increased muscle tension. Steeneken and Hansen (1999) stated that the increased muscle tension of the vocal folds and the vocal tract may adversely affect the speech quality. This assumption could be affirmed by the work of Mendoza and Carballo (1998) showing an increase in jitter and shimmer in conditions of experimentally induced stress by cognitive tasks. In contrast, other studies reported a decrease in jitter and shimmer as tasks become cognitively demanding (Keränen et al., 2004; Rothkrantz et al., 2004). Two further voice-related parameters were investigated in the past: Yap et al. (2015) reported a decrease in the harmonics-to-noise ratio and an increase in the cepstral peak prominence when high levels of cognitive load are induced by time pressure; the results exhibited statistically significant differences.

Previous studies in literature also investigated the effects of cognitive load on prosodic features. It has been shown that an increase in cognitive load is reflected by an increase in the fundamental frequency (Mendoza and Carballo, 1998; Scherer et al., 2002; Rothkrantz et al., 2004; Boril et al., 2010; Huttunen et al., 2011a), whereas only a few studies showed no consistent effects (Tolkmitt and Scherer, 1986; Lively et al., 1993). Additionally, some works showed a decrease in the variability of the fundamental frequency under high cognitive load conditions (Lively et al., 1993; Huttunen et al., 2011a). An increased cognitive load is also linked to an increase in the intensity (Rothkrantz et al., 2004; Huttunen et al., 2011a) and a decrease in energy decay (Scherer et al., 2002).

Features that characterise the human vocal tract have also been found to be indicative of cognitive load. Boril et al. (2010) reported an increase in the first and fourth formant centre frequencies in voiced speech segments, but no significant effects in the second and third. A reduction in the second formant has been shown

Table 2.5 Review of the effects of cognitive load on low-level speech features

Feature	Tendency
Fundamental frequency	+
Fundamental frequency variability	−
Intensity	+
Centre frequency of formant F_1	0, +
Centre frequency of formant F_2	0, −
Centre frequency of formant F_3	0
Centre frequency of formant F_4	+
Spectral centre of gravity	+
Jitter	−, +
Shimmer	−, +
Harmonics-to-noise ratio	−
Cepstral peak prominence	+

+: feature value increases as cognitive load increases,
−: feature value decreases as cognitive load increases,
0: no effect

by Yap et al. (2011a). However, Lively et al. (1993) reported no effects of various workload levels on the first three formants.

Spectral properties of the speech signal have been found to reflect the cognitive load experienced by the speaker. Scherer et al. (2002) demonstrated that high cognitive load leads to a decrease in the energy below 500 Hz compared to the energy in the range of 500 to 1600 Hz. Apart from that, an increase in the spectral centre of gravity and spectral energy spread could be found in Boril et al. (2010). Tolkmitt and Scherer (1986) reported that significant changes in the spectral energy distribution due to cognitive stress can only be observed in female subjects.

2.4 Cognitive Load Applications

Apart from the formulation of theoretical concepts and models regarding the construct cognitive load, practical aspects were also considered over the years. Assuming that cognitive load is automatically assessed, possibly in real-time, many applications could benefit from measurements of the individual mental state. For instance, cognitive load measurement can be used in sports (Ruiz et al., 2010) and medical research (Marchese et al., 2003). Adaptive interfaces can benefit from cognitive state detection and by that optimise human-machine interaction (Grootjen et al., 2006, 2007; De Greef et al., 2007). Cognitive load

measurement can be deployed in environments in which speech is used as part of the work using the phone or face-to-face communication (Khawaja et al., 2014), for example, in bushfire management centres, air traffic control rooms, and call centres. Cognitive load monitoring can also be helpful for developing better shared mental models to enhance team performance (Fan and Yen, 2007; Fan et al., 2010).

The following sections focus on relevant areas where the concept of cognitive load can be applied, namely education in Section 2.4.1, user interface design in Section 2.4.2, in-vehicle systems in Section 2.4.3, and aviation in Section 2.4.4.

2.4.1 Education

In the context of CLT (Section 2.1.3), high levels of cognitive load may exceed the capacity of working memory and possibly hamper the construction of schemas during learning. Substantial empirical evidence for instructional guidelines of the CLT framework came from a series of experiments and meta-analyses (cf. Sweller et al., 2011). With this in mind, monitoring a user's state is crucial to optimise learning materials, control the learning process, select the learning tasks, and thus improve the overall learning efficiency.

Anvari et al. (2013) demonstrated how to identify students, who are talented in 3D computer graphics programming, by conducting a variety of tests including cognitive load measurement based on task and performance. As a consequence, students can be supported so that talented students benefit from receiving advanced training, whereas less talented students can be given extra tutoring. Harms (2013) reported that personalised tutorials can be helpful to assist programmers in learning new programming skills—the effectiveness of automatically generated tutorials can be improved by monitoring the user's intrinsic cognitive load with respect to programming expertise, whereby extraneous load can be reduced by selecting suitable programming concepts.

Intelligent tutoring systems can benefit from cognitive load classification based on speech. In this context, Zhang et al. (2006) proposed a system to teach children mathematics and physics using an active learning task, where children play with Lego gears while interacting with the system through speech by answering questions (e.g., 'Why is the smaller gear stronger?').

Gillmor et al. (2015) examined the effects of cognitive load on students' performance in a mathematics test by using different representations of mathematics assessment items. The authors confirmed that cognitive load can be reduced by signalling important information, aesthetic item organisation, and removing extraneous content, which, in turn, improves student performance. CLT is also applied to various settings such as classroom, workplace, and self-directed learning for medical education (Young et al., 2014).

Coyne et al. (2009) discussed how augmented cognition and neuroergonomics can be expanded into training. In this connection, they used EEG and ocular data in order to classify the user state in real-time and proposed an adaptive system, which is able to improve learning by adjusting the training difficulty. The corresponding model of this approach combines elements from Wickens' *multiple resource theory* (Wickens, 2008) and Sweller's *cognitive load theory* (Sweller et al., 2011).

2.4.2 User Interface Design

In order to provide intuitive and effective human-computer interfaces, which are also aimed for efficient task completion, user interface and information system designers should consider human cognitive processes and its implications. Some suggestions deserve to be considered in interface design. This concerns particularly effects of split-attention and modality (Sweller et al., 1998). The former effect can be addressed by physical integration of disparate sources of information in the design in order to avoid load for mentally integrating sources of information that have been separated either spatially or temporally. Regarding modality, audiovisual representations of textual and pictorial material may deliver the information in a more effective way.

The authors Tracy and Albers (2006) applied the principles of CLT (Section 2.1.3) to web site design. Thus, the designer can consider what factors are causing high cognitive load and then redesign the problem areas (e.g., poor navigation or cryptic categories) to reduce the load. Moreover, Schultz et al. (2007) proposed a usability test system to verify user interfaces of web applications based on the user's emotions and cognitive load from physiological data.

With regard to information retrieval, Hu et al. (1999) found that interface design may have a significant effect on system-user communication, regardless of users' familiarity with the search task, and that a graphical user interface may be more effective in supporting such communication than a list-based design. The aggregate cognitive load of list-based interfaces was comparable to that of graphical ones—the difference was not statistically significant. On the other hand, subjects exhibited a higher level of satisfaction with a graphical interface than with a list-based interface. Both cognitive load and the level of satisfaction were measured using self-reporting methods.

Shi et al. (2009) proposed a multimodal interface, called CAMI (Cognition-Adaptive Multimodal Interface). It considers cognitive states and was primarily designed for a large metropolitan traffic incident and emergency management system. Different components have been integrated, such as speech analysis, galvanic skin response, handwriting on a tablet monitor, and hand-gesture based

interaction. As part of this interface, the system determines when and what information is delivered to the user (operator) in which way.

2.4.3 In-Vehicle Systems

Interactive in-vehicle systems become more and more popular. These systems comprise infotainment systems, telematics systems, and portable systems, such as mobile phones. The driver's capability to control a vehicle may be influenced by secondary in-vehicle activities, which, in turn, could have negative safety consequences. Reliable automatic cognitive load detection can contribute to the design of safety systems and other intelligent in-vehicle interfaces.

Some work has been done in order to optimise spoken dialogue systems based on cognitive load measurements of the driver. For instance, Kun et al. (2011) investigated the effects of different dialogue behaviours on the driver's cognitive load using pupil diameter. They revealed that the rise of the pupil diameter is related to increased cognitive activity when he or she is attempting to find the word described by the other conversant. In the work of Boril et al. (2010), speech production differences of the driver were explored by two secondary cognitive tasks: interaction with the co-driver (low cognitive load) and calls to automated spoken dialogue systems (high cognitive load). The outcome was that various speech parameters, such as tempo in speech and the fundamental frequency, are suitable for cognitive load discrimination.

Engström et al. (2005) reported results from simulated and real motorway driving concerning the effects of cognitive load on driving performance. It was found that cognitive load does not affect speed and results in reduced lane keeping variation as well as increased gaze concentration towards the road centre. These results are attributed to the driver's perception of increased risk and the need to increase the safety. On the other hand, visual load (e.g., operating the radio) leads to reduced speed and increased lane keeping variation. Moreover, Crundall and Underwood (1998) studied the differences between novices and experienced drivers using different levels of cognitive load imposed by different types of roads. Measures of eye movements were taken, which indicated that experienced drivers select visual strategies according to the complexity of the roadway, whereas strategies of novices are too inflexible to meet the requirements.

There is also ongoing research regarding user experience while driving (e.g., Jung et al., 2011). Hess et al. (2013) stated that the perceived experience is not only influenced by the actual product, system, service, or object, but also by contextual factors, such as emotions and cognitive load. In order to investigate the relationship between cognitive load and user experience in a driving scenario, they introduced two driving simulator environments to conduct experiments.

2.4.4 Aviation

Monitoring the mental state can also be helpful in civil and military aviation because cognitive overload of pilots and crew directly affects safety.

Huttunen et al. (2011a,b) showed correlations between speech-based parameters and cognitive load according to situation awareness, information processing, and decision making of pilots in a simulator task. The results of both studies indicate that speech features can be used to monitor the speaker state and support pilot training in a simulator environment. Based on real-time cognitive load monitoring of pilots, assistance could be requested in time before critical situations occur. Svensson and Wilson (2002) analysed the effects of task (mission) complexity on mental state and performance of pilots using the measurement techniques eye fixation rate, heart rate, and blink rate. The authors found significant relationships between heart rate and rated workload, mental capacity, situational awareness, and performance.

Military pilots have to function in environments that unavoidably have an effect on speech production and reception. Keränen et al. (2004) analysed changes in the production of speech by military aircraft pilots. Speech of a total of 19 pilots was recorded, both on real flights and in a simulator—there were clear increases in the intensity and fundamental frequency of the speech signal for all the pilots in more intense combat situations. Hence, both prosodic parameters could potentially serve as indicators of situation demands and the need for support.

The importance of crew communication on the flight deck is discussed in the work of Sexton and Helmreich (2000). Results of linguistic analyses showed that the way language is used is correlated with the workload level during flight and, therefore, with individual performance and error rates.

Chapter 3

Speech Material

Over the last couple of years, various tasks have been used to investigate the limitations of the human working memory in conjunction with speech parameters, for example, reading comprehension (Yin et al., 2007), Stroop tests (Yap et al., 2010b), arithmetic tasks (Gorovoy et al., 2010), digit span tasks (Quatieri et al., 2015), and driving under cognitive load (Boril et al., 2010). However, speech corpora within this field are rather rare and in general not freely available for scientific research. Besides the level of cognitive load, the way the cognitive load is induced may also influence the manner in which users behave. One can assume that such factors are reflected in different speech parameters. In order to explore these aspects and to evaluate cognitive load assessment systems, speech data is needed. In this chapter, two databases are presented that are used for the investigations in this thesis, namely *CLSE* and *CoLoSS*.

The CLSE database is presented in Section 3.1. It includes speech recordings of subjects who participated in three different tasks: Stroop test with time pressure, Stroop test with dual-task, and a reading span task. Each task is composed of three levels of cognitive load. The partitioned form of the database—a version of the data collection used for the INTERSPEECH 2014 Computational Paralinguistics Challenge—was provided for this thesis.

Cognitive load monitoring, which is optimised for environments in which users are involved in the process of acquiring new or modifying existing knowledge, may enrich application-oriented systems that are aimed at assisting users in enhancing their learning efficiency. So far, there are no known speech-based corpora available that focus on cognitive load in the context of learning and, secondly, that provide a more sensitive approach for the assessment of cognitive load than that of the straightforward classification problem. Therefore, a new cognitive load corpus, called CoLoSS, has been developed, which includes speech recordings of subjects who performed a learning task. The speech material is numerically described by labels that reflect the amount of cognitive resources devoted during the learning task procedure. CoLoSS is one of the major contributions of this thesis. Details are found in Section 3.2.

3.1 CLSE Database

The *Cognitive Load with Speech and EGG* (CLSE) database was developed by Yap (2012) with the goal to investigate the effects of cognitive load on speech, especially on glottal parameters. Therefore, in addition to pure acoustic speech signals recorded using a microphone, the database also includes electroglottograph (EGG) signals. The cognitive load tasks employed in this database and the data collection are described in the following sections.

3.1.1 Task Design

The CLSE database contains speech recordings of participants who performed two variants of the Stroop test and a reading span task. The total recording session for each participant lasted about one hour. For each task, three different levels of cognitive load were induced: low, medium, and high. All tasks were presented to the participants using a desktop computer.

3.1.1.1 Two Variants of the Stroop Test: Time Pressure and Dual-Task

The Stroop test (Stroop, 1935) is a psychological test, which aims at investigating the effects of interference and the related reaction (or processing) time of a person. More specifically, subjects are required to name the font colour of a word while the colour name is different. For instance, if the displayed font colour is blue and the colour name is 'red', then the correct answer is 'blue'. This condition is generally more demanding than a task in which the word meaning and the font colour are not incongruent. The Stroop test is often used to study psychophysiology responses to mental stress (e.g., Salahuddin et al., 2007; Karthikeyan et al., 2014; Vanitha et al., 2017).

Two variants of the Stroop test are employed in the CLSE database corresponding to the way the cognitive load is induced: Stroop test with time pressure (CLSE-Time) and Stroop test with dual-task (CLSE-Dual). For both variants, ten different colours were used (black, blue, brown, grey, green, orange, pink, purple, red, and yellow). For each trial, 20 colour names were displayed in random order, i.e., a particular colour name appeared twice. Table 3.1 gives an overview of the task designs used for CLSE-Time and CLSE-Dual. In the low cognitive load conditions, font colours correspond to the colour name, while they are incongruent in the medium and high cognitive load condition.

In the low and medium cognitive load condition of CLSE-Time, participants were required to read a sequence of words displayed at the same time. In these cases, task completion is conducted self-paced. The high cognitive load level was

Table 3.1 Task design of CLSE-Time and CLSE-Dual

Cognitive load	CLSE-Time	CLSE-Dual
Low	Font colour = colour name, word sequence is displayed at the same time (self-paced)	Font colour = colour name, 1.0 s display interval for word sequence
Medium	Font colour ≠ colour name, word sequence is displayed at the same time (self-paced)	Font colour ≠ colour name, 1.0 s display interval for word sequence
High	Font colour ≠ colour name, 0.8 s display interval for word sequence	Font colour ≠ colour name, 1.0 s display interval for word sequence, and tone counting

induced by time pressure—each word of the word sequence, one after another, was displayed on the screen within a short time span of 0.8 s.

Regarding all cognitive load levels of CLSE-Dual, each word of the word sequence was displayed at a time interval of 1 s. In the high cognitive load condition, participants were required to perform a tone-counting task (without using fingers) while naming font colours. Either a low-pitched (1 kHz) or high-pitched tone (2 kHz) was played through headphones in two-second intervals, 200 ms before the next word appeared. At the end of each trial, the total number of high-pitched tones had to be named.

For both Stroop test variants, trials were repeated three times for each load and subjective ratings were collected at the end of each trial. The subjective ratings exhibit statistically significant differences in terms of cognitive load conditions (cf. Yap, 2012).

3.1.1.2 Reading Span Task

The reading span task required participants to read aloud a series of short sentences while memorising a sequence of letters. This type of task design is commonly used to measure the working memory capacity by the maximum number of items that can be stored (Daneman and Carpenter, 1980).

This task is divided into sets of trials, whereby each trial is structured as follows: Participants are asked to read a displayed sentence, which is either semantically logical or illogical. For instance, the sentence ‘I like to walk in the sky’ is semantically illogical. The sentence is then verified by the participant with the statement ‘true’ or ‘false’. Subsequently, a single letter, which has to be stored in working memory, is presented on the screen for approximately 0.8 s.

The number of possible letters is twelve (‘F’, ‘H’, ‘J’, ‘K’, ‘L’, ‘N’, ‘P’, ‘Q’, ‘R’, ‘S’, ‘T’, and ‘Y’).

A set of trials varied from two to five sentences. At the end of each set of trials, the participants were required to recall the memorised letters in the correct order. For instance, a set of three trials corresponds to a number of three sentences as well as three letters. After recalling the letters, subjects were asked to rate their invested mental effort on a scale from 1 to 9. In total, 21 sets of trials (5×2 trials, 5×3 trials, 5×4 trials, and 6×5 trials) were conducted resulting in 75 produced utterances per subject. These sets were presented in random order, i.e., a participant did not know beforehand how many sentences and letters have to be processed in a sequence.

Cognitive load labelling for the reading span task is based on the assumption that as the number of letters increases, the amount of working memory utilised for the task will also increase. Consequently, the cognitive load labels were assigned by means of the number of sentences: low—reading the first sentence (no letter); medium—reading the second sentence (first letter); high—reading the third sentence (second letter), fourth sentence (third letter), and fifth sentence (fourth letter).

3.1.2 Data Description and Partitioning

The CLSE database includes speech from 26 native English speaking students (20 male and 6 female)—it was recorded in Sydney (Australia). Each of the students participated in three tasks (Section 3.1.1.1 and 3.1.1.2). The data collection is grouped into the corresponding sub-databases *CLSE-Time* (Stroop test with time pressure), *CLSE-Dual* (Stroop test with dual-task), and *CLSE-Span* (reading span task).

Speech was recorded using a close-talk microphone at a sampling frequency of 48 kHz and a 16 bit resolution. The speech signals were afterwards downsampled to 16 kHz. Each instance of the database corresponds to a task trial, i.e., an audio file comprises speech of 20 named colours in the case of CLSE-Time and CLSE-Dual, while a single sentence is given per audio file in CLSE-Span. Further details are shown in Table 3.2. All instances of the entire database are associated with a class label, which corresponds to a particular cognitive load level: low, medium, or high. CLSE-Time and CLSE-Dual are provided with data that include three instances per cognitive load level for each subject. In contrast, an unequal distribution among classes can be observed in the case of CLSE-Span; per subject, 21 instances are available for low as well as medium and 33 instances for high cognitive load.

The database also includes the uttered single letters of the reading span task and electroglottograph (EGG) data concerning all tasks recorded simulta-

Table 3.2 Data description of the CLSE database

Description	CLSE-Time	CLSE-Dual	CLSE-Span
Number of subjects	26	26	26
Number of instances	234	234	1,950
Average duration per instance [s]	16.5	20.8	4.0
Total duration [hh:mm]	01:04	01:21	02:09

Table 3.3 Partitioning of the CLSE database

#	Train	Devel	Test	$\sum$
Cognitive load				
Low	297	189	216	702
Medium	297	189	216	702
High	429	273	312	1,014
Task				
CLSE-Time	99	63	72	234
CLSE-Dual	99	63	72	234
CLSE-Span	825	525	600	1,950
$\sum$	1,023	651	744	2,418

Sum of instances are shown for each cognitive load level and each cognitive load task. Abbreviations: Devel (development set), Test (test set), Train (training set)

neously with the speech. Furthermore, data for background model training is included (CLSE-Story) comprising approximately 80 s of read speech per subject. However, recordings of the single letters, EGG data, and CLSE-Story were not considered in this thesis.

As part of the INTERSPEECH 2014 Computational Paralinguistics Challenge (ComParE) (Schuller et al., 2014), the Cognitive Load Sub-Challenge has been established. The goal was to provide a test-bed for the automatic recognition of speakers' cognitive load as a three-class problem. For this purpose, the CLSE database was partitioned into speaker disjoint subsets for training, development, and testing. The training set contains 11 subjects (9 males and 2 females), the development set 7 subjects (5 males and 2 females), and the test set 8 subjects (6 males and 2 females). Table 3.3 summarises the partitioning of the database for the Cognitive Load Sub-Challenge. The challenge evaluation refers to the entire database, i.e., results obtained on all three task-test sets are considered for the final score. The partitioned version of the CLSE database, as given in Table 3.3, was provided for the investigations in this thesis.

3.2 CoLoSS Corpus

Although much effort has been invested in collecting and segmenting speech data in recent years, there is still a lack of publicly available annotated databases in the field of cognitive load, in particular in the context of learning. Moreover, automatic speech-based cognitive load recognition modelled so far was only considered to be a classification problem, where the level of task difficulty is used for data labelling.

As a contribution to this research area, a new speech-based corpus has been developed by the author of this thesis, called CoLoSS (***Co****gnitive* ***Lo****ad by* ***S****peech and performance data in a* ***S****ymbol-digit dual-task*). This corpus represents a subset of data collected in conjunction with previous studies (Wirzberger et al., 2017, 2018) in which the task design (Section 3.2.1) and cognitive load indicators (Section 3.2.2) were defined. The recordings of the previously collected data have been refined within the scope of this thesis (Section 3.2.3) for evaluation purposes in the field of automatic speech-based cognitive load recognition. Compared to existing data collections that aim at evaluating automatic cognitive load recognition from speech, the CoLoSS corpus differs in two key aspects: (1) it focuses on cognitive load induced by learning processes; (2) it provides continuous numerical labels as a reference for cognitive load.

The corpus material will be made freely available for research purposes (Herms et al., 2018). In the following sections, details on the corpus are given including the task design, recording conditions, cognitive load indicators, and statistics.

3.2.1 Task Design

The fundamental object of this task design is to assess the residual cognitive resources of subjects while they are performing a learning task. To accomplish this, the dual-task methodology was applied: a visual-motor primary task in which subjects are required to learn the assignment of a symbol combination to a single symbol while simultaneously memorising a sequence of five different digits from an auditory-verbal secondary task. The symbol assignments in the primary task reflect knowledge schemas that have to be formed across the trials.[1] Subjects' performance in the secondary task accordingly can be considered to be the reference for cognitive load associated with the primary task.

[1] The following four different abstract geometric symbols were used: circle, triangle, square, and star. The assignment of a symbol combination to a particular symbol was not necessarily logical. This ensures that prior knowledge of subjects was excluded and, consequently, they had to form their own learning strategy.

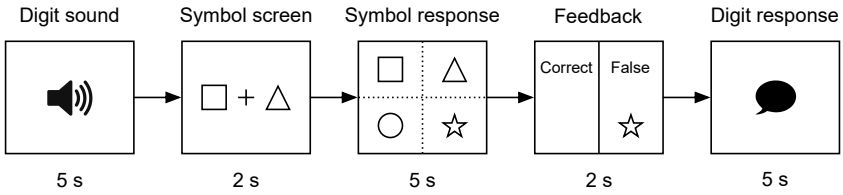


Figure 3.1 Schematic representation of a dual-task trial applied for the CoLoSS corpus. Different screens and the corresponding durations in seconds (s) are shown. Primary task: step two to three; secondary task: step one and five

In total, the task comprises a set of 64 trials and was presented to the participants using a desktop computer. Participants were guided in each trial by different screens as illustrated in Figure 3.1. Between two screens, except between the first two screens, a blank white screen was displayed with a duration of 200 ms. A single trial is composed of the following five steps: (1) Digit sound: hearing a sequence of five different digits in the range of 1 to 9 (in random order) generated by a text-to-speech system for German; (2) Symbol screen: seeing one out of four randomly generated combinations of different, abstract geometrical symbols, where the order of the symbols must be considered; (3) Symbol response: selecting one out of four possible symbols in a randomly arranged 2×2 grid via mouse click; (4) Feedback: obtaining feedback, accompanied by the correct symbol in the case of false response to foster correct schema acquisition; (5) Digit response: verbally recalling the five-digit sequence of step 1 in correct order.

It is obvious that symbol combinations of the first trials were presented for the first time and symbol response in step 3 constitutes a chance of 25%. The task difficulties varied between subjects, but not within the task, by the number of symbols displayed on the screen in step 2. A distinction was made between an easy and a difficult condition by two and three symbols, respectively.

With reference to the cognitive load theory (Section 2.1.3), this task design is associated to various assumptions: Intrinsic cognitive load is represented by the characteristics of the primary task, i.e., number and order of symbols. The extraneous cognitive load is represented by the secondary task requirements. Finally, the overall load, including germane cognitive load, is reflected in the subjects' performance concerning the secondary task.

3.2.2 Chosen Measures

In order to obtain a highly sensitive measure of the subjects' performance for the primary and secondary task (Section 3.2.1), an efficiency score was computed

using the *likelihood model* approach proposed by Hoffman and Schraw (2010).[2] The calculation is based upon the ratio between performance and effort, whereby performance P is represented by the accuracy of problem solving and effort is represented by the time T required. With this in mind, efficiency can be expressed as (Hoffman and Schraw, 2010):

$$\mathrm{Eff} = \frac{P}{T}, \tag{3.1}$$

where T is unequal to zero. In the ongoing, the efficiency of the primary task and the secondary task are denoted as $\mathrm{Eff_{PT}}$ and $\mathrm{Eff_{ST}}$, respectively.

For $\mathrm{Eff_{PT}}$ calculation, the performance was determined by the symbol response correctness in the *symbol response* stage, while the reaction time needed to select a symbol constitutes the effort component. Note, reaction time refers to the visual stimulus regarding the appearance of the 2×2 grid.

The performance component of $\mathrm{Eff_{ST}}$ was defined as the correctness of the subject's response regarding the five-digit sequence in the *digit response* stage. Inspired by evaluation approaches for automatic speech recognition systems, the word accuracy was chosen as a measure of performance. That is, the spoken words were aligned against the correct words in the reference and, subsequently, the number of substituted words S, deleted words D, and inserted words I were determined. The word accuracy (WA) was then calculated by considering the word error rate (WER) as follows (Lee, 1988, p. 147):

$$\mathrm{WER} = \frac{S + D + I}{N}, \tag{3.2a}$$

$$\mathrm{WA} = 1 - \mathrm{WER}, \tag{3.2b}$$

where N is the number of words in the reference. Indeed, word error rates can be greater than 100%, which may occur if $I > D$. It was decided to avoid negative accuracy values by setting the lower limit to zero. Since the reference contains five words, one of the following six values (here, not expressed as a percentage) could be obtained by the parameter WA: 0, 0.2, 0.4, 0.6, 0.8, or 1.

The effort component of $\mathrm{Eff_{ST}}$ was determined by the time starting from the presentation of the visual stimulus (speech bubble in the *digit response* stage) to the end of the last uttered digit, hereinafter referred to as the 'verbal response duration' (VRD). In fact, VRD includes the onset latency, which reflects the reaction time from the stimulus to the onset of the first uttered digit. This time span involves complex cognitive processing for mentally representing the message, selecting words, and retrieving syntactic as well as phonetic properties

[2] Although efficiency measures have so far been calculated using a single or the primary task, there is no reason why they should not be calculated using the secondary task (Sweller et al., 2011, p. 80).

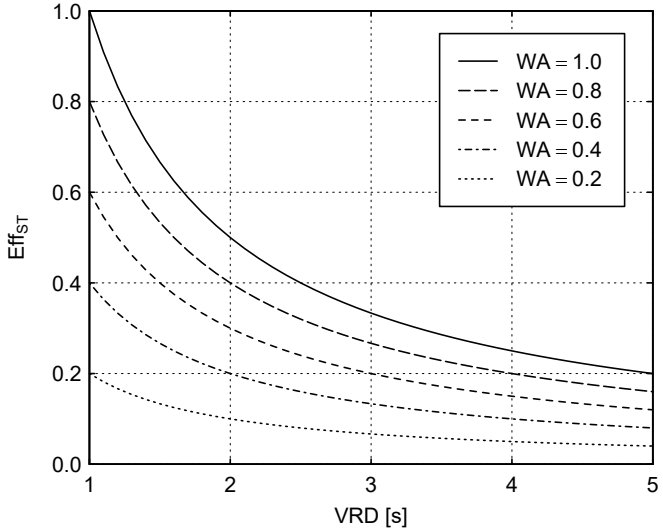


Figure 3.2 Relationship between word accuracy (WA), verbal response duration (VRD), and secondary task efficiency (Eff_{ST})

and, moreover, motor processing for articulation is required. Although the reaction time is assumed to reflect how cognitive demanding the primary task was (Sweller et al., 2011, p. 78), there is more information that can be derived from the response in the secondary task. This assumption is attributed to the characteristics of the task itself, namely the occurrence of many digits to be named instead of only a single reaction. Consequently, as stated above, the complete response including the onset latency was taken into account to determine the effort component.

A theoretical perspective of the resulting Eff_{ST} curves is given in Figure 3.2. The efficiency score ranges from 0 to 1 while the VRD in seconds ranges from 1 to 5.[3] The highest score ($\text{Eff}_{\text{ST}} = 1$) can only be achieved by WA $= 1$ along with VRD ≤ 1 seconds. The Eff_{ST} is highly sensitive for VRD between 1 and 2 seconds. With 2 seconds and more, the score is reduced to 0.5 and below. This allows considering the reaction time (onset latency) in a sensitive manner for utterances that are equal in duration.[4]

[3]Note that VRD can also be shorter than 1 second. This may occur if only one or two digits are named with a short reaction time. Again, this is only a theoretical perspective.

[4]Here, the duration of an utterance is defined as the time from the onset of the first digit to the end of the last digit.

3.2.3 Recording and Post-processing

In total, 123 German-speaking students from the Technische Universität Chemnitz (Germany) participated in the task. Speech was recorded using a clip-on microphone connected to a pocket transmitter-receiver system (Sennheiser ew 112). Mono signals were captured at a sampling frequency of 48 kHz and a 24 bit resolution using a mobile recording device (Roland R-88). Each recording session refers to a particular subject and covers the entire task across the 64 trials. A recording session lasted about 20 minutes. The segments concerning the uttered five-digit sequence in the *digit response* stage (Section 3.2.1) were afterwards extracted using the time-codes (5 seconds + 0.5 seconds tolerance) from the task log-data.

Unfortunately, the data of 28 subjects had to be excluded, because some of them did not fulfil the requirements of the entrance test conducted using an operation span (OSPAN) task[5], or did not confirm the license agreement for providing the data, while in other cases, subjects with lack in language proficiency (non-native speakers) could be observed. Furthermore, some audio segments of the remaining subjects were excluded due to manifold reasons: a segment contains only silence; a segment does not include at least one uttered digit; a segment contains disturbing noise while the subject is speaking, for example, caused by unintended gesticulation. To obtain enough data per subject for the investigations, it was decided that only subjects with at least 75% of valid segments are included in the speech corpus.

In order to determine the verbal response duration (Section 3.2.2) in a precise manner, the audio segments were annotated by two student assistants using time markers in the software Audacity (Audacity Team, 2012). Note, this annotation process involves omitting any sound including uttered content after the end of the last uttered digit. Subsequently, all duration values were double checked by another student assistant.

The efficiency score of the secondary task constitutes a promising indicator for cognitive load and can be used for data labelling (Section 3.2.4). However, since the audio segments of the corpus were theoretically contaminated with information on the verbal response duration and, thus, partly with information on the secondary task efficiency, segments were further processed by automatic trimming. Figure 3.3 illustrates an original audio segment including annotations and the resulting speech segment after trimming. The trimming procedure used

[5]The OSPAN task was used to obtain a baseline for the participants' individual working memory capacity. The task consists of three sets, each provided with five trials. For each trial, participants were required to remember presented letters from the Latin alphabet while evaluating the correctness of math problems. At the end of each trial, participants had to determine the correct order of letters. At least 85% accuracy on the math problems was required.

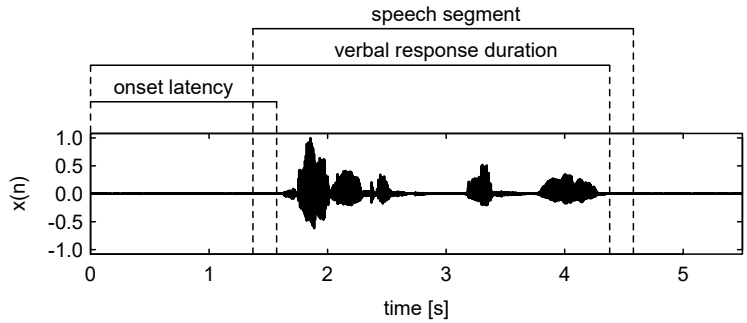


Figure 3.3 Onset latency, verbal response duration and segment of a recorded speech signal

works as follows: Energy threshold-based activity detection (Section 4.1.5) is applied to the audio signal in order to obtain the onset of the first activity and the end of the last activity. Here, an activity refers to any sound, which can be caused by speech, breathing, filled pauses, lip-smacking, and so forth. A tolerance of 200 milliseconds (ms) is then added to the segmental boundaries of the activity detector guaranteeing that information on speaker's activity is not lost due to detection errors. If the length of the tolerance exceeds the limits of the original audio segment, as much silence as needed is added to fill the 200 ms at the beginning and/or the end. Afterwards, the trimmed audio segments are transcoded to 16 kHz with a 16 bit resolution in WAV—the audio file format of the corpus.

3.2.4 Cognitive Load Labels

The principle behind the dual-task method is that the available working memory capacity, which is not allocated in connection with the primary task, can be employed for a second task. Therefore, performance measures of the secondary task, such as the number of correct answers, number of errors, and/or time required, can be used as a reliable and valid reference for cognitive load associated with the primary task (cf. Section 2.2.1). Regarding the task design described in Section 3.2.1, the work of Wirzberger et al. (2017) backed up the hypothesis that as learning progresses along with the sequence of trials, both the primary task efficiency and secondary task efficiency increase. With this in mind, the variable of interest for data labelling is the secondary task efficiency (Section 3.2.2), which considers performance (word accuracy) as well as effort (time required). This label assignment serves the purpose of describing the audio data

with the following assumed relationship: Secondary task efficiency reflects the amount of the speaker's devoted cognitive resources while he or she is performing the secondary task. The higher the load imposed by cognitive, learning processes in terms of the primary task, the lower the efficiency score of the secondary task.

A second version of the cognitive load labels was realised by applying unsupervised discretisation. More specifically, the continuous numerical values of the secondary task efficiency were converted to categorical values by equal-width binning. This method divides the range of values x into N intervals with $width = (max_x - min_x)/N$. As a consequence, each interval has the same width. Here, as in the CLSE database (Section 3.1), a ternary classification problem is of interest, i.e., the chosen number of intervals was set to $N = 3$. It is worth noting that this method is an experimental approach for another representation of the labels. Other discretisation methods such as equal-frequency binning or supervised discretisation can also be used. The optimal conversion of a continuous numerical indicator into cognitive load classes constitutes an open issue for future work.

3.2.5 Data Description

Table 3.4 gives an overview of the data collection. The corpus includes 70 native speakers of German, whereby 18 are male and 52 are female (9 male and 26 female per task difficulty). Due to the exclusion of some audio files (cf. Section 3.2.3), the number of instances varies across the speakers (min $= 48$, max $= 64$, $\mu = 58.23$, $\sigma = 18.41$). Compared to the overall CLSE database (Section 3.1), the CoLoSS corpus includes 44 more different speakers and 1,658 more instances. In the following, the corpus material is given at a glance:

- Audio files (WAV, mono, 16 kHz, 16 bit) containing German speech (digits) from secondary task trials,
- Subject id, trial id, and information about the gender,
- Information on primary task conditions (easy and difficult),
- Primary task performance measurements (symbol response correctness, reaction time, efficiency),
- Secondary task performance measurements (word accuracy, verbal response duration, efficiency),
- Cognitive load labels (secondary task efficiency as continuous numerical and ordered categorical variables).

Figure 3.4 illustrates the progression of primary task and secondary task efficiency over trials. One can see an increasing trend by means of simple linear regression, but only a slight difference between conditions. For a deeper insight into data, linear mixed-effects models with standardised (z-score) predictors were

Table 3.4 Data description of the CoLoSS corpus

Description	Condition		
	easy	difficult	all
Number of subjects	35	35	70
Number of instances	2,050	2,026	4,076
Average duration per instance [s]	2.66	2.67	2.66
Total duration [hh:mm]	01:31	01:30	03:01

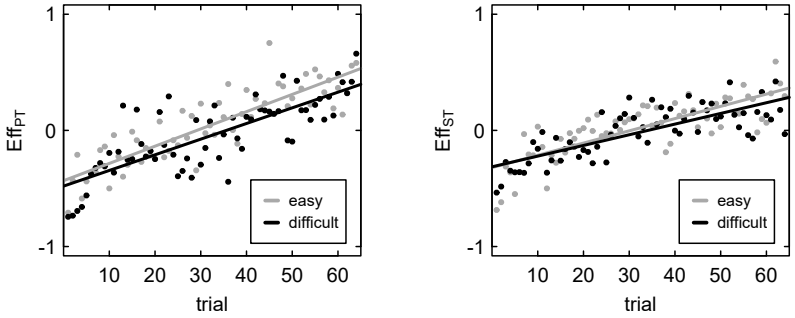


Figure 3.4 Standardised primary task efficiency (left) and secondary task efficiency (right) over trials averaged across subjects for each task condition (easy and difficult). The relationship between trial and efficiency is shown by simple linear regression

used to consider individual effects; the parameters of interest are: condition, the interaction between trial and condition, and efficiency across trials per subject. Results confirm an increase in efficiency for the primary task ($\beta = 0.273$, $p < 0.001$, RMSE $= 1.003$, $R^2 = 0.178$) as well as for the secondary task ($\beta = 0.186$, $p < 0.001$, RMSE $= 0.983$, $R^2 = 0.478$).[6] For both tasks, there is no significant difference between the easy and the difficult condition ($p > 0.05$).[7] Hence, for the investigations in this thesis, no distinction is made between these conditions.

Since the secondary task efficiency (Eff_{ST}) is suggested to be the reference for cognitive load (cf. Section 3.2.4), the underlying parameters are discussed in

[6]For RMSE computation, a leave-one-subject-out cross-validation was used. R^2 was obtained by a pseudo R^2 calculation based on the linear mixed-effects models.

[7]Subjective measures based on the questionnaire of Leppink et al. (2013) show similar results. For both intrinsic load and extraneous load, no differences between conditions can be observed. Regarding germane load, significantly higher ratings ($p < 0.05$) are obtained by the easy condition.

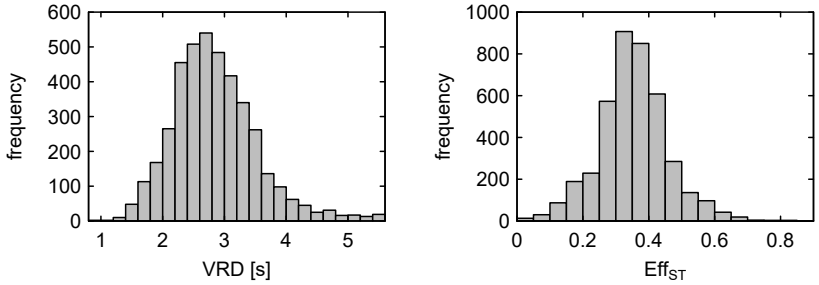


Figure 3.5 Histograms for the verbal response duration (left) and secondary task efficiency (right)

more detail. Based on Equation (3.1), $\mathrm{Eff}_{\mathrm{ST}}$ is calculated by the ratio between word accuracy (WA) and verbal response duration (VRD). Regarding the word accuracy, the five-digit sequence is error-free in almost all cases with a frequency of 3,480. The remaining accuracies show comparably very low frequencies: 2, 23, 63, 332, and 176 for WA = 0, 0.2, 0.4, 0.6, and 0.8, respectively. By taking all WA values into account, the mean is 0.94 and the standard deviation is 0.15. This affirms that the secondary task is rather simple so that it did not tend to distract subjects from working on the primary task. The distributions of VRD and $\mathrm{Eff}_{\mathrm{ST}}$ are illustrated in Figure 3.5. The data of VRD ($\min = 0.83$, $\max = 5.5$, $\mu = 2.85$, $\sigma = 0.68$) as well as $\mathrm{Eff}_{\mathrm{ST}}$ ($\min = 0$, $\max = 0.86$, $\mu = 0.35$, $\sigma = 0.11$) are not normally distributed, tested using the Shapiro-Wilk test ($p < 0.05$). For the VRD data, the skewness is 0.81 and the kurtosis is 1.41 indicating that the distribution is slightly skewed to the right and has a higher, sharper peak than the normal distribution. Similar properties can be observed for $\mathrm{Eff}_{\mathrm{ST}}$, where the skewness is 0.21 and the kurtosis is 1.15.

As stated in Section 3.2.4, the corpus was complemented with categorical cognitive load labels obtained by applying unsupervised discretisation to the $\mathrm{Eff}_{\mathrm{ST}}$ data. With respect to the assumptions regarding $\mathrm{Eff}_{\mathrm{ST}}$ and by involving all conditions, the discretisation method used leads to the following function:

$$\mathrm{CL}(\mathrm{Eff}_{\mathrm{ST}}) = \begin{cases} L_1 & \text{for} \quad 0.58 < \mathrm{Eff}_{\mathrm{ST}} \leq 0.86 \\ L_2 & \text{for} \quad 0.29 < \mathrm{Eff}_{\mathrm{ST}} < 0.58 \\ L_3 & \text{for} \quad 0.00 \leq \mathrm{Eff}_{\mathrm{ST}} < 0.29, \end{cases} \tag{3.3}$$

where L_1, L_2, and L_3 represent low, medium, and high cognitive load (CL), respectively.[8] Due to the statistical characteristics of $\mathrm{Eff}_{\mathrm{ST}}$, the resulting distri-

[8]For clarity reasons, the ranges of $\mathrm{Eff}_{\mathrm{ST}}$ values shown are rounded to two decimals.

bution among classes is highly unbalanced (L_1: 109, L_2: 3,051, and L_3: 916). Hence, for conducting classification experiments, it is strongly recommended to apply resampling techniques (Section 6.1.1) before model training is performed.

Chapter 4

Speech Processing

Speech information contained in audio is usually represented in a compact form by specific parameters (i.e., features) that describe the characteristics of the signal at a comparatively lower rate. This chapter gives an overview of methods for digital speech processing and feature extraction. First, the fundamentals of audio signal processing (Section 4.1) are introduced. This is followed by a description of those features that are investigated in the following chapters of this thesis. The extracted features cover typical features from the cognitive load literature (Section 2.3.2) and address adjacent fields such as speech under stress. Since the speech material used (Chapter 3) does not provide a basis for linguistically motivated features, these are not taken into account. In this thesis, the term 'feature type' is used to differentiate between features regarding the extraction method applied to speech. Feature types that describe similar aspects in speech are summarised into a 'feature group'. Four different groups are considered: prosodic features (Section 4.2), spectral features (Section 4.3), voice quality features (Section 4.4), and Teager energy operator based features (Section 4.5).

Almost all feature types presented in this chapter can be classified as low-level features. The only exception refers to the duration in speech (Section 4.2.3)—a feature type for representing the prosodic characteristics. There are some arguments indicating that the implemented duration-based features may be considered to be high-level: (1) they can be interpreted by humans in an intuitive manner (e.g., speaking rate or number of pauses); (2) from a technical point of view, the basis for feature computation is provided by different representations of speech (e.g., phonemes or segment length), not by operating on the audio signal directly.

The features of the above-mentioned groups can be processed further. Section 4.6 describes how contextual information is derived on frame-level and how features are summarised for machine learning purposes. At the end of this chapter (Section 4.7), the implemented overall process chain for feature extraction is presented.

4.1 Fundamentals

The following sections provide the reader with fundamentals of audio signal processing. Basically, signals convey information that is contained in patterns of variation. For technical reasons, discrete-time signals are required, which are representations of sampled continuous-time signals. Broadly speaking, sampling is the transformation of a continuous signal into a sequence of numerical values. Discrete-time signals are mathematically denoted by the function $x(n)$, where n is an integer between $-\infty$ and ∞ (Oppenheim et al., 1999, p. 9). Signal discretisation can be achieved by using an analogue-to-digital (A/D) converter. For the perfect reconstruction of the original continuous-time signal, the *Nyquist-Shannon* sampling theorem requires that the sampling frequency must be at least twice the highest frequency in the signal (Oppenheim et al., 1999, p. 146). For more details regarding signal representation and transformation, the reader is referred to Oppenheim et al. (1999).

In the remainder of this thesis—for reasons of simplicity—the notation $x(n)$ refers to a value at index n on any level of discrete data representations.

4.1.1 Short-Time Analysis

Speech is a non-stationary signal, i.e., it has varying properties over time (Hermansky, 1999, p. 118). In order to capture the information in specific regions of speech, the signal needs to be segmented into shorter units. In audio signal processing, a sliding short-duration window is typically applied to the discrete signal along the time axis (Jayan, 2016, p. 22). The region of the window is then considered to be stationary, meaning that its statistical properties are constant. The signal within the boundaries of the window can be extracted by multiplying the value of the discrete-time signal $x(n)$ at time n by the value of the window $w(n)$ at time n. The extracted signal from a window is called a frame. Consequently, applying a sliding window along the time axis results in a sequence of frames (Jurafsky and Martin, 2009, p. 331).

There exist various types of weighting window functions; the choice of a suitable function depends on the feature to be determined. A rectangular window passes the original signal form, which is sufficient for the analysis in the time domain (Schuller, 2013, p. 44). However, a rectangular window cuts off the signal at its boundaries resulting in discontinuities, which, in turn, can lead to undesirable effects concerning time-to-frequency transformation (Jurafsky and Martin, 2009, p. 331). A compromise is the use of window functions that are characterised by a *soft* fade in and fade out in terms of time and, thus, also in terms of frequency (e.g., Hamming, Hanning, and Gaussian window) (Fellbaum, 2012, p. 77).

Two important parameters influence the results of the short-time analysis: the window length and the frame rate. The window length corresponds to the number of samples to be processed at a time. It is crucial to define an adequate length. Short-duration windows result in localised values, whereas long-duration windows yield averaged measurements (Jayan, 2016, p. 24). The parameter frame rate has the function of how frequently the measurement is performed on the input signal. In slow signal variations, a lower frame rate is sufficient, whereas fast signal variations require a higher frame rate. The frame rate is typically parameterised by a frame-shift value controlling the time-shift in the window placement (Jayan, 2016, p. 25). Depending on the window length, the frame rate also controls the degree of overlap between successive windows.

4.1.2 Spectrum and Cepstrum

Some speech features require the transformation of a windowed signal into the frequency domain. The tool for extracting spectral information from a discrete-time signal $x(n)$ is commonly known as the discrete Fourier transform (DFT). The output of the DFT is a complex number $X(k)$ representing the magnitude and phase of a frequency component k in the original signal. The DFT at time n and frequency k is given as (Jurafsky and Martin, 2009, p. 333):

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}kn}. \tag{4.1}$$

The DFT is typically computed using the efficient algorithm fast Fourier transform (FFT). It is worth noting that this implementation of the DFT only works for values of N that are powers of 2.

Additionally, useful information from the signal can be derived by the cepstrum. The cepstrum is computed by taking the log for each amplitude value in the magnitude spectrum and, subsequently, performing the back-transformation to the time domain. More formally, the cepstrum is defined as the inverse DFT of the log magnitude of the DFT of a signal, written as (Jurafsky and Martin, 2009, p. 335):

$$c(n) = \sum_{n=0}^{N-1} \log(|X(k)|) e^{j\frac{2\pi}{N}kn}. \tag{4.2}$$

The term ‘cepstrum’ is formed by reordering the first letters of the term ‘spectrum’. The term ‘quefrency’—formed in a similar way based on the term ‘frequency’—is commonly used as axis label in order to emphasise the difference between time domain and cepstral domain (Lerch, 2012, p. 102).

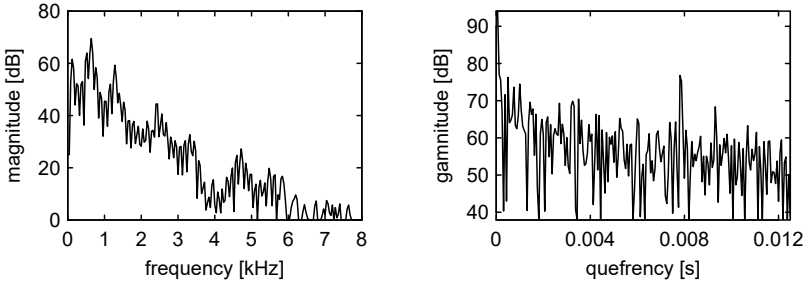


Figure 4.1 Spectrum (left) and cepstrum (right) of a 25 ms voiced speech frame

As described in Section 2.3.1.2, the voiced speech waveform is generated by a source signal (of a particular fundamental frequency) which is passed through the vocal tract filter. Figure 4.1 illustrates the spectrum and the cepstrum of a 25 ms frame extracted from a voiced speech signal. The cepstrum provides a way of separating the source and the filter: a large peak corresponds to the fundamental frequency; components modelled by the vocal tract filter are concentrated in the low quefrency region (Havelock et al., 2008, p. 476). Thus, the cepstrum can be used for the detection of the fundamental frequency and the extraction of features for speech recognition.

4.1.3 Autocorrelation Function

Correlation functions measure the degree of similarity between two signals at a particular time lag. The autocorrelation function (ACF) is a special case, which measures how well the input signal matches a time-shifted version of itself. This type of correlation function is a useful tool for short-time speech analysis. For instance, it can be applied to determine the periodicity in signals (cf. Section 4.2.2). The short-time autocorrelation function of a discrete-time signal $x(n)$ at lag τ is defined as (Harrington and Cassidy, 2012, p. 146):

$$ACF(\tau) = \sum_{n=0}^{N-1-\tau} x(n)x(n+\tau), \tag{4.3}$$

where N refers to the length of the window. Equation (4.3) indicates that $ACF(\tau)$ is computed at a given lag value relative to the start of the window at $n = 0$. The maximum value of $ACF(\tau)$ is at lag $\tau = 0$ (i.e., no time-shift) because there is no difference between the data points to be multiplied (Lerch, 2012, p. 27).

4.1.4 Contour Smoothing

The contours of low-level speech features may include undesirable short-term fluctuations or detection errors that can affect the results of further processing steps. A common way to reduce noise in a sequence of data points is to apply the simple moving average (SMA) filter. The SMA filter smooths a data contour by averaging the values within a moving window. In equation form, the output of the filter at frame n is given by (Smith, 2013, p. 277):

$$SMA(n) = \frac{1}{W} \sum_{i=-(W-1)/2}^{(W-1)/2} x(n+i), \tag{4.4}$$

where W represents the window length, i.e., the number of frames to be processed. The parameter W assumes odd numbers due to the centring of n within the window (symmetrical averaging).

The SMA filter has been implemented for this thesis and is applied to each low-level feature type. A window length of $W = 3$ is used in order to retain most of the information regarding short-term trends.

4.1.5 Activity Detection

In audio analysis, not all frames of the signal are of interest. Some parts of the signal can contain silence or disturbing noise that may not be important for the analysis of speech. This concerns in particular automatic speech recognition (ASR) systems: the recognition performance of ASR is—among other factors—related to the position of segmental boundaries regarding spoken units (Droppo and Acero, 2008, p. 662).

Voice activity detection (VAD), also known as speech activity detection, aims to discriminate between speech and non-speech in audio signals. Early approaches for VAD are based on energy threshold (Wilpon et al., 1984; Woo et al., 2000), zero-crossing rate (Hahn and Park, 1992), periodicity measures (Tucker, 1992), linear predictive coding measures (Rabiner and Sambur, 1977), and cepstral features (Haigh and Mason, 1993). More recently, data-driven approaches are followed by using features for classifier training to discriminate between speech and non-speech in noisy environments. For instance, Gaussian mixture models (Misra, 2012), support vector machines (Enqing et al., 2002), recurrent neural networks (Eyben et al., 2013b), deep belief networks (Zhang and Wu, 2013), and C-means clustering (Górriz et al., 2006) have been successfully employed.

For this thesis, a threshold-based activity detector has been implemented. This component is decoupled from specific data sources, meaning that it can be

used for experimental purposes. Let $x(n)$ be the sequence of frames obtained by feature extraction and let $a(n)$ be the sequence of activity frames, where each frame is initialised as 'inactivity'. The following four parameters can be adjusted and work in concert:

1. *Activity threshold*: The status 'activity' is set in $a(n)$ at those indices where $x(n)$ yields a value above the threshold.
2. *Frame tolerance*: In $a(n)$, a defined number of frames are set to 'activity' before and after each detection (from step 1).
3. *Gap threshold*: The gap between two 'activity' frames in $a(n)$ is filled with 'activity' if the number of frames between them is below the threshold.
4. *Series threshold*: A series of consecutive 'activity' frames in $a(n)$ are set to 'inactivity' if the number of these frames is below the threshold.

This activity detector is applied for this thesis in order to determine different types of speech events (Section 4.2.3.2) and to post-process all low-level feature types in the way that only voiced speech is considered based on the segmental boundaries of the fundamental frequency (Section 4.2.2). It should be noted that this type of activity detector only works robustly in non-noisy environments. Real-world conditions require methods that are more sophisticated.

4.2 Prosodic Features

Broadly speaking, prosody refers to the melody in speech. Prosody is characterised by variations of the parameters intensity, fundamental frequency, and duration. In combination, these parameters contribute to the production and perception of rhythm, lexical stress, lexical tone, tempo, and intonation of an utterance (Fletcher, 2010, p. 521).

The following sections describe the extraction of the *intensity* (Section 4.2.1), the *fundamental frequency* (Section 4.2.2), and *duration*-based features (Section 4.2.3).

4.2.1 Intensity

The *intensity* is a physical, measurable parameter of a sound based on the amplitudes over time. In contrast, the term 'loudness' refers to a perceptual quantity, which can only be measured on the basis of human observers (Lerch, 2012, p. 71). The degree of intensity depends on recording conditions, i.e., it decreases as the distance from the sound source decreases.

The energy of an audio signal can be used as an approximation to measure the intensity. The signal energy of a discrete-time signal $x(n)$ with a finite number

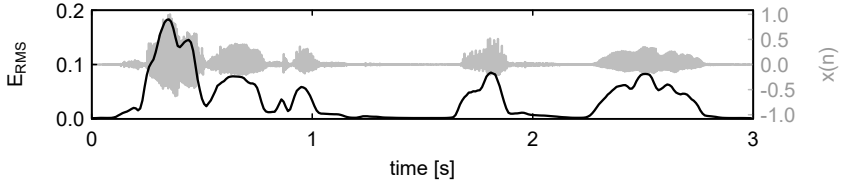


Figure 4.2 Waveform (background) and intensity (foreground) of a speech signal

of samples is defined by the formula (Manolakis and Ingle, 2011, p. 25):

$$E = \sum_{n=0}^{N-1} x(n)^2. \tag{4.5}$$

Alternatively, the root mean square (RMS) related to the amplitudes (i.e., RMS energy) can be computed, which is one of the most common intensity features. The RMS energy of a discrete-time signal $x(n)$ is calculated by (Lerch, 2012, p. 71):

$$E_{RMS} = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} x(n)^2}. \tag{4.6}$$

For this thesis, the RMS energy of a signal is extracted by using the analysis tool Praat (Boersma, 2002). A window length of 25 ms is chosen with a frame-shift of 10 ms. Figure 4.2 illustrates the corresponding contour extracted from a speech signal. The result of each frame is in the range of $0 \leq E_{RMS} \leq 1$. The value will be zero if the processed part of the signal is silence.

With reference to previous studies in the scientific literature (Section 2.3.2.3), the intensity of a signal increases as the level of cognitive load increases. This relationship is particularly true for the data of CLSE-Time (cf. Figure A.1 in Appendix), whereby statistically significant differences are given between low and medium as well as low and high cognitive load. For CLSE-Dual, an increase can be observed from low to medium load with a significant difference, but there is a drop from medium to high cognitive load (not significant). However, the difference between low and high cognitive load in CLSE-Dual is still statistically significant. In the case of CLSE-Span, there is a decrease from low to medium load and an increase from medium to high load; only the former exhibit a statistically significant difference. Regarding the distribution of intensity in the CoLoSS corpus, no significant differences between cognitive load levels can be observed.

4.2.2 Fundamental Frequency

The *fundamental frequency* F_0 of a speech signal represents the frequency of the vocal fold vibration. Hence, this frequency component appears when speech is voiced. This type of feature can be regarded as the acoustic equivalent to the perceptual unit pitch. Figure 4.3 shows the fundamental frequency for a speech signal. There are various detection techniques to obtain this frequency in both the time domain and the frequency domain. An excellent overview of techniques is given by Gerhard (2003).

For this thesis, the fundamental frequency is determined in the time domain by the straightforward autocorrelation approach using the analysis tool Praat (Boersma, 1993, 2002). The measurement interval is set to 10 ms at a window length of 40 ms. As pointed out in Section 4.1.3, the autocorrelation function $ACF(\tau)$ has a global maximum at the time lag $\tau = 0$, because the signal is identical to itself. If there are global maxima outside $\tau = 0$, then the signal is called periodic and there exists a time lag T_0, which is called the period. The fundamental frequency F_0 of a periodic signal is defined as (Lerch, 2012, p. 8):

$$F_0 = \frac{1}{T_0}. \tag{4.7}$$

The correlation of a periodic waveform to itself decreases to a minimum when the time lag increases to half of the period and increases back to a maximum when the time lag increases to the length of one period. In the case of harmonically complex waveforms there may be no global maxima outside $\tau = 0$, but possibly a number of local maxima. If there is a local maximum which is large enough, then the signal has a periodic part. The first large peak of the autocorrelation function indicates the fundamental period of the waveform.

Previous works revealed that the fundamental frequency F_0 increases along with cognitive load (cf. Section 2.3.2.3). This can only be partially confirmed in this thesis (cf. Figure A.1 in Appendix). For both CLSE-Time and CLSE-Dual, F_0 increases from low to medium cognitive load with a statistically significant difference but decreases slightly from medium to high load. However, for CLSE-Dual, the difference between low and high cognitive load is still significant. For the CoLoSS corpus, F_0 values indicate a tendency to a slight increase from medium to high cognitive load—the difference is significant. On the other hand, confidence intervals of F_0 overlap between low and medium as well as low and high cognitive load. Opposite effects can be observed for CLSE-Span: F_0 decreases as cognitive load increases; a significant separation between low and medium load can be observed, but there are almost no changes between medium and high cognitive load.

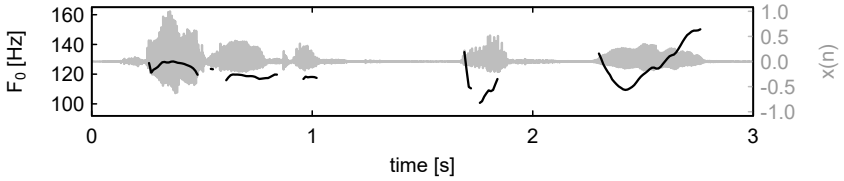


Figure 4.3 Waveform (background) and fundamental frequency (foreground) of a speech signal

4.2.3 Duration

Duration-based features refer to the temporal aspects in speech. This type of prosodic analysis implies the detection of spoken units (e.g., voiced segments or phonemes) because the timing of substructures can reveal tempo and pausing in speech. From a technical point of view, this raises two kinds of issues: (1) How to obtain spoken units in an automatic manner? (2) Which features can be derived from these spoken units?

Some contributions were made to this, especially with the focus on syllables. For instance, Mermelstein (1975) proposed a segmentation algorithm based on the minimum of loudness to detect the boundaries of syllables. In the work of Hunt (1993), syllables are detected by applying recurrent neural networks with energy and cepstral features as input parameters. In De Jong and Wempe (2007), the intensity and fundamental frequency are used in connection with thresholds to restrict the speech signal; the remaining intensity peaks are considered to be syllable locations. These locations are then used for computing the speaking rate by the number of syllables per second. The authors Verhasselt and Martens (1996) presented an automatic speech detector based on a multilayer perceptron to determine phone boundaries and the speaking rate (number of phones per second).

Although some systems seem to be controversial due to their error rates, the work of Mirghafori et al. (1996) showed the usefulness of measuring tempo based on hypothesised phones of an automatic speech recognition (ASR) system. Moreover, previous works demonstrated the importance of ASR-based features for various use-case scenarios including the prediction of cognitive load (e.g., Montacié and Caraty, 2014; Quatieri et al., 2015), Parkinson's disease (e.g., Zlotnik et al., 2015) as well as deception and sincerity (e.g., Herms, 2016). For further information on automatic speech recognition, the reader is referred to Young et al. (2006).

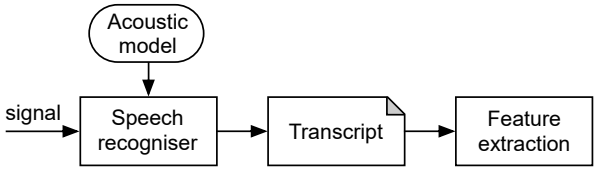


Figure 4.4 ASR-based feature extraction

4.2.3.1 Phoneme-Level Features

The speech recognition approach is followed in this thesis to extract features that refer to the temporal properties of speech. In order to achieve this, an ASR-based feature extractor has been implemented by the author of this thesis; a preliminary version was already successfully employed in Herms (2016). The process chain of the system is illustrated in Figure 4.4. It consists of two main stages, namely speech recognition and feature extraction.

The open-source framework CMU Sphinx-4 (Walker et al., 2004) has been used for implementing the speech recognition component. The input data of the speech recogniser comprise the audio data (e.g., from a WAV file) and an acoustic model, while the output is a transcript including phonemes and the corresponding timecodes in milliseconds. In order to detect the phoneme sequences for the English language (i.e., CLSE database—Section 3.1), the pretrained US English generic acoustic model *cmusphinx-en-us-5.2*[1] has been used, which is provided by CMU (Carnegie Mellon University). For the German language (i.e., CoLoSS—Section 3.2), a separate acoustic model has been trained based on the 'German open source corpus for distant speech recognition' (Radeck-Arneth et al., 2015) in conjunction with a pronunciation lexicon, which has been created using the grapheme-to-phoneme conversion tool (Reichel, 2012) of the BAS (Bavarian Archive for Speech Signals) web services (Kisler et al., 2016). The resulting acoustic model for German consists of context-dependent triphone Hidden Markov Models with 32 Gaussians per state. Table 4.1 gives an overview of the symbolic units for each acoustic model. Since all datasets used in this thesis contain utterances that are not characterised as spontaneous speech, the filled pauses are not considered for feature computation.

Feature extraction is performed on the basis of the generated transcripts. A single transcript refers to an instance (audio file) of a particular speech corpus. Each transcript is considered from the first to the last phoneme. As further indicators, consonants and vowels are detected in the phoneme sequences. Next,

[1] https://sourceforge.net/projects/cmusphinx/files/, accessed 19 March 2015.

Table 4.1 List of symbolic units for English and German produced by ASR

Type	English (ARPAbet notation)	German (SAMPA notation)
Phonemes	*AA*, *AE*, *AH*, *AO*, *AW*, *AY*, *B*, *CH*, *D*, *DH*, *EH*, *ER*, *EY*, *F*, *G*, *HH*, *IH*, *IY*, *JH*, *K*, *L*, *M*, *N*, *NG*, *OW*, *OY*, *P*, *R*, *S*, *SH*, *T*, *TH*, *UH*, *UW*, *V*, *W*, *Y*, *Z*, *ZH*	*2:*, *6*, *9*, *?*, *@*, *C*, *E*, *E:*, *I*, *N*, *O*, *OY*, *S*, *U*, *Y*, *Z*, *a*, *a:*, *aI*, *aU*, *b*, *d*, *e:*, *f*, *g*, *h*, *i:*, *j*, *k*, *l*, *m*, *n*, *o:*, *p*, *r*, *s*, *t*, *u:*, *v*, *x*, *y:*, *z*
Silent pauses	*SIL*	*SIL*
Filled pauses	*BREATH*, *COUGH*, *NOISE*, *SMACK*, *UH*, *UM*	

syllable patterns are derived based on the concept of the consonant-vowel structure as described in Farinas and Pellegrino (2001). In this context, a syllable pattern is defined as $C^n V$, where n is an integer that may be zero, consecutive vowels are merged, and clusters without vowels are discarded. For instance, the sequence *CCVV.CCV.CV.CCCV.CV.CCC* results in the following five valid syllables: *CCV*, *CCV*, *CV*, *CCCV*, and *CV*.

The phonemes, silent pauses, and derived syllables are considered for the extraction of the following ten static features:

- *Utterance duration*: time code of the first phoneme subtracted from the time code of the last phoneme,
- *Silent pause duration*: the total duration of silent pauses,
- *Silent pause frequency*: the total number of silent pauses,
- *Mean silent pause duration*: silent pause duration divided by the silent pause frequency,
- *Silent pause duration ratio*: silent pause duration divided by the utterance duration,
- *Silent pause frequency ratio*: silent pause frequency divided by the total number of phonemes,
- *Speaking rate (phoneme-based)*: the total number of phonemes divided by the utterance duration,
- *Speaking rate (syllable-based)*: the total number of syllables divided by the utterance duration,
- *Articulation rate (phoneme-based)*: the total number of phonemes divided by the utterance duration excluding silent pause duration,
- *Articulation rate (syllable-based)*: the total number of syllables divided by the utterance duration excluding silent pause duration.

Temporal variation is obtained by taking each single spoken unit into account. The following four different duration types are extracted:

- *Phoneme durations*,
- *Consonant durations*,
- *Vowel durations*,
- *Syllable durations*.

A precise description of tempo in speech can be determined by considering short-time variations. For this purpose, a sliding window with a length of 500 ms and a shift of 100 ms is applied along the time axis. The number of spoken units is then captured for each short-time interval. Two speaking rate contours are extracted:

- *Speaking rate contour (phoneme-based)*,
- *Speaking rate contour (syllable-based)*.

Means and 95% confidence intervals for a selection of duration-based features under different levels of cognitive load are found in Appendix (Figure A.1). The articulation rate in the case of CLSE-Dual and CoLoSS shows effects similar to those given in the scientific literature (cf. Section 2.3.2). More precisely, for CLSE-Dual, there is a monotonically decreasing trend across the levels of cognitive load with statistically significant differences. The same applies to CoLoSS, with the exception that the difference between medium and high cognitive load is insignificant. Interestingly, an opposite effect can be observed for CLSE-Time, whereby the articulation rate significantly increases when a high cognitive load is induced (by time pressure). Obviously, compared to the articulation rate, the trend of the mean consonant as well as vowel durations turned to the opposite direction, although significant effects are only found in the case of CLSE-Time and CoLoSS. Both corpora exhibit an increase in terms of the consonant duration from low to medium and low to high cognitive load. Looking at the mean vowel duration, the results between databases differ considerably: CLSE-Time shows a significant increase from low to medium and a significant decrease from medium to high cognitive load; for CoLoSS, there is a monotonically increasing trend with statically significant differences between low and high as well as medium and high cognitive load. It is generally accepted that longer pauses in speech are indicative of cognitive load. The same conclusion can be drawn with regard to CLSE-Time where a significant increase from low to medium and low to high cognitive load can be observed for silent pause duration. Similar effects are obtained by CoLoSS; values exhibit statistically significant differences between low and high as well as medium and high cognitive load. In contrast, CLSE-Dual shows a significant decrease in silent pause duration from low to medium cognitive load.

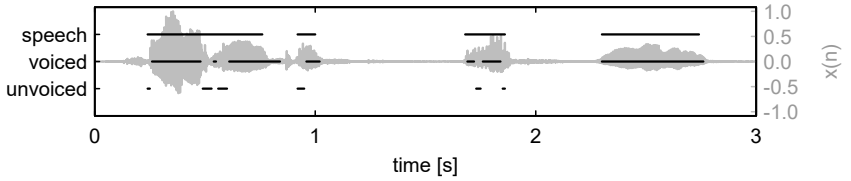


Figure 4.5 Waveform (background) and segment types (foreground) of a speech signal

4.2.3.2 Segments and Onset Latency

The features presented in this section measure the duration of various speech events based on low-level features. Audio activity detection (Section 4.1.5) is first applied to a feature contour and the length of each detected segment is then computed by the difference between its last and first frame. The following three different types of segment lengths in milliseconds are extracted:

- *Speech segment lengths*,
- *Voiced segment lengths*,
- *Unvoiced segment lengths*.

Figure 4.5 illustrates the three segment types for a speech signal. Speech segments are detected based on the intensity (Section 4.2.1) of an audio signal. For voiced segments, the boundaries of the fundamental frequency contours (Section 4.2.2) are used directly as a reference. Unvoiced segments are then obtained by the difference between the speech segments and the voiced segments. Moreover, speech corpora may include stimuli information realised as starting points in audio files. In order to take the reaction time of a speaker into account, the *onset latency* is extracted by considering the timecode of the first frame contained in the first speech segment.

4.3 Spectral Features

Spectral features play an important role in the recognition of the speaker's mental state because they convey the frequency content of the speech signal and provide complementary information to prosodic features. This group of features (including cepstral features) is based on the short-time spectral analysis.

Three spectral feature types are considered in this thesis. First, the *spectral centroid* (Section 4.3.1) is described. The spectral centroid is associated with the brightness of a sound. This is followed by the well-known *Mel Frequency Cepstral Coefficients* (MFCCs, Section 4.3.2)—the standard feature in automatic

speech recognition systems. MFCCs describe the spectral properties of a signal in a compact form with only a few parameters. The third feature of this group refers to the resonance frequencies of the human vocal tract, also known as the *formants* (Section 4.3.3).

In this thesis, spectral features are extracted using the analysis tool Praat (Boersma, 2002). For short-time spectral analysis, the window length is set to 25 ms with a frame-shift of 10 ms. Praat applies a Gaussian window function for MFCCs and formants by default. For the spectral centroid, the Hamming window function is used.

4.3.1 Spectral Centroid

The *spectral centroid* (SC) refers to the centre of gravity of spectral energy. It is defined as the weighted sum of the power spectrum divided by its unweighted sum (Lerch, 2012, p. 45):

$$SC = \frac{\sum_{k=0}^{N-1} k \cdot X(k)^2}{\sum_{k=0}^{N-1} X(k)^2}, \tag{4.8}$$

where $X(k)$ is the magnitude corresponding to frequency k, and N is half of the sampling frequency (Nyquist frequency).

The spectral centroid is well correlated with the timbre in terms of brightness. For instance, low values indicate more low-frequency content, which results in low brightness, whereas a brighter sound dominates in higher frequency (Park, 2009, p. 399). Figure 4.6 illustrates the spectral centroid contour for a speech signal. It can be seen that the spectral centroid spikes at transients between speech and non-speech frames; it is high during pauses (outside the figure).

The spectral centroid is commonly applied in the field of musical genre classification (e.g., Tzanetakis and Cook, 2002) or musical instrument recognition (e.g., Eronen and Klapuri, 2000). Moreover, this feature has been used in speech emotion recognition (e.g., Schuller and Rigoll, 2006; Chen et al., 2012). In this thesis, the entire frequency band of a speech signal is considered for spectral centroid determination. There are also other approaches such as the computation of spectral centroid frequencies and spectral centroid amplitudes within frequency subbands (e.g., Cummins et al., 2011; Le et al., 2011).

Analyses of the mean spectral centroid reveal that there is an increasing trend as the level of cognitive load increases from low to medium (cf. Figure A.3 in Appendix), whereby differences concerning CLSE-Time as well as CLSE-Dual are statistically significant. For both corpora, differences are negligible between medium and high load, but spectral centroid values are still significantly higher in the most intense cognitive load condition than in the lowest. An opposite

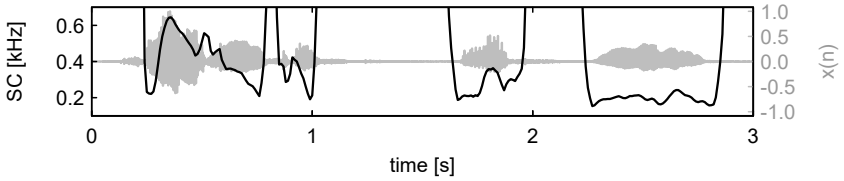


Figure 4.6 Waveform (background) and spectral centroid (foreground) of a speech signal

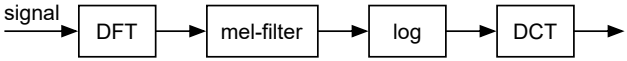


Figure 4.7 MFCC feature extraction

effect is given by CoLoSS: the lowest mean spectral centroid is obtained by the highest level of cognitive load; the difference to medium load is significant. For CLSE-Span, no significant effects can be observed.

4.3.2 Mel Frequency Cepstral Coefficients

The *Mel Frequency Cepstral Coefficients* (MFCCs) can be regarded as a compact description of the spectral properties of an audio signal. The MFCC feature extraction procedure is closely related to the computation of the cepstrum (Section 4.1.2).

The computation of MFCCs requires four basic steps as depicted in Figure 4.7. In the first step, the frequency spectrum of a windowed signal is determined by the discrete Fourier transform (DFT). The next step is to apply a bank of overlapping triangular bandpass filters that follow the mel-warped spectrum. To this end, a non-linear frequency scale—the mel scale—is used to model the non-linear human perception of pitch (Lerch, 2012, p. 51). Consequently, MFCCs are frequently considered to be a perceptual feature. There are differences between MFCC implementations in terms of the mel-warped spectrum. A common definition of the mel-frequency scale is (Lerch, 2012, p. 80):

$$mel(f) = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right). \tag{4.9}$$

The mel-spectrum is obtained by multiplying the magnitude spectrum of the Fourier transformed signal by each of the triangular mel weighting filters. Sub-

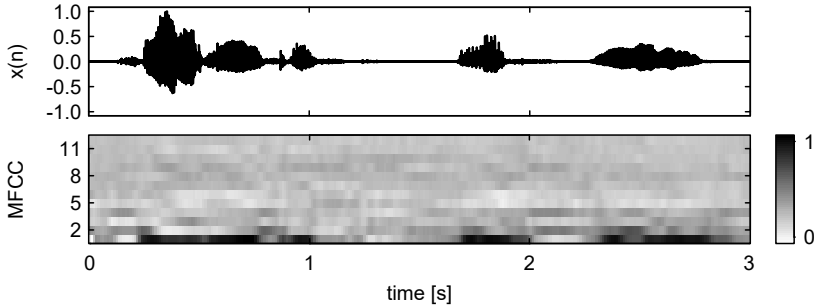


Figure 4.8 Waveform and MFCC heat map of a speech signal

sequently, the logarithm of the spectrum is taken—it has been found that the perceived loudness of a signal is approximately logarithmic (Kim et al., 2006, p. 81). Finally, the discrete cosine transform (DCT) is applied to the logarithm of the mel-filter outputs. The result is a set of cepstral coefficients. In summary, the computation of MFCCs can be expressed as (Davis and Mermelstein, 1980, p. 199):

$$MFCC(n) = \sum_{k=1}^{K} \log\left(X_{mel}(k)\right) \cdot \cos\left(n \cdot \left(k - \frac{1}{2}\right) \frac{\pi}{K}\right), \quad n = 1, 2, \ldots, M, \tag{4.10}$$

where $X_{mel}(k)$ represents the mel-spectrum with the energy output of the kth filter, and M is the number of cepstrum coefficients.

Typically, MFCCs are computed for a number of coefficients that is less than the number of mel-filters. For this thesis, the first 12 coefficients of a 26 mel-filter bank specification are used. Figure 4.8 illustrates these 12 MFCCs for a speech signal. Each row in the heat map corresponds to a coefficient; darker shades of grey represent higher (normalised) values.

Correlation measurements between MFCCs and cognitive load are found in Appendix (Figure A.5). For the mean MFCCs, almost all values show a very low relationship to cognitive load across different tasks. However, concerning CLSE-Time, a small negative relationship can be observed for MFCC 1, 2, and 9; the results are significant at a level of 0.01. Regarding the variation of MFCCs, there is only little if any relationship across all tasks. The only exception is given by MFCC 11 in the case of CLSE-Time, which exhibits a low positive correlation ($p < 0.01$).

4.3.3 Formants

Formants are resonance frequencies of the human vocal tract. Since different vowels are produced by different configurations of the vocal tract, they are reflected in different kinds of resonances (Jurafsky and Martin, 2009, p. 273). Although formant frequencies vary with the articulation of speech, they are also characterised by the size and proportion of speech organs of an individual speaker (Eriksson, 2012, p. 50). Formants are often directly visible in spectral representations. They can be described in terms of the centre frequency, amplitude, and bandwidth. Figure 4.9 illustrates a 300 ms segment of a voiced speech signal; the high energy portions in the frequency domain are represented by darker shades of grey and the corresponding first three formant contours are represented by lines.

A common approach to estimate formants involves searching for peaks in spectral representations. However, there are some difficulties in formant determination, because the short-time spectra may contain vocal tract's resonance frequencies that are too close to each other or some dominant frequencies exceed resonance frequencies in amplitude. A simplified structure of the speech spectrum is needed, which can be achieved by linear predictive coding (LPC). The LPC method provides an accurate estimate of speech parameters that relate to the configuration of the vocal tract. It derives a compact representation of the spectral characteristics of a signal. The basic idea of linear prediction is that the current speech sample $\hat{x}(n)$ can be estimated by a linear combination of previous output samples of the original signal $x(n)$, expressed as follows (Lerch, 2012, p. 29):

$$\hat{x}(n) = \sum_{i=1}^{P} a_i \cdot x(n-i). \tag{4.11}$$

Equation (4.11) indicates that $x(n-i)$ are the preceding values, a_i are the predictor coefficients, and P is the order of the predictor. The prediction error is

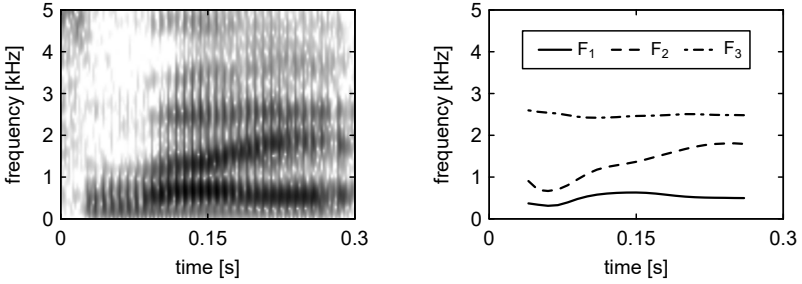


Figure 4.9 Spectrogram (left) and the first three formants (right) of a speech signal

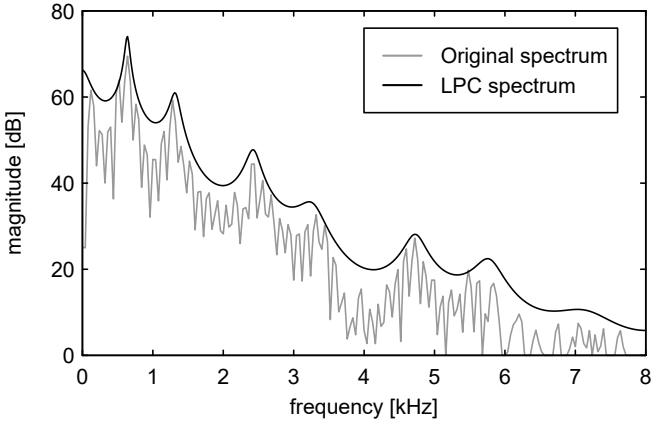


Figure 4.10 Fourier and LPC spectrum of a 25 ms voiced speech frame

computed by the difference between the real output and the prediction (Lerch, 2012, p. 29):

$$e(n) = x(n) - \hat{x}(n). \tag{4.12}$$

The predictor coefficients a_i are usually estimated by minimising the mean squared prediction error. The minimum can be found by setting the differential $\partial e^2(n)/\partial a_i$ equal to zero. This leads to a system of P linear equations—solving these equations yields the predictor coefficients a_i. Finally, the LPC spectrum is obtained by applying the DFT to the LPC coefficients. For the mathematics behind LPC analysis and formant estimation, see Schuller (2013).

For this thesis, the centre frequency f and bandwidth b of the first three formants, denoted hereinafter by $F_{1\text{–}3}(f,b)$, are extracted using the analysis tool Praat (Boersma, 2002). For each analysis window, Praat computes LPC coefficients and determines the formant values. Figure 4.10 illustrates the LPC and the Fourier spectrum of a 25 ms voiced speech frame. The corresponding first three formant centre frequencies are 641 Hz, 1313 Hz, and 2422 Hz.

The utilisation of formant frequencies for cognitive load assessment seems to be controversial due to inconsistent research findings (cf. Section 2.3.2.3). The same applies to the results regarding the speech material used in this thesis (cf. Figure A.4 in Appendix). Most effects can be observed for the CLSE-Time dataset. The centre frequency of F_1 shows a monotonically increasing trend across all cognitive load levels with statistically significant differences between low and medium as well as low and high cognitive load. Regarding the F_3 centre frequency, a significant decrease can be observed from medium to high

load. Both F_2 and F_3 bandwidths decrease as cognitive load increases; more precisely, there is a significant difference between low and high cognitive load for both measures and, additionally, the bandwidth of F_3 exhibits a significant difference between medium and high load. For the data of CLSE-Dual, only the centre of the first formant frequency shows a significant effect, namely a decrease from medium to high cognitive load; yet, there is no monotonic trend across the levels of cognitive load. In the case of CLSE-Span, there is a significant increase concerning the centre frequency of F_2 from medium to high cognitive load and a significant decrease in the centre frequency of F_3 from low to high cognitive load. No significant effects are given by the CoLoSS corpus.

4.4 Voice Quality Features

Voice quality features are related to the characteristics of the source signal in speech. As described in Section 2.3.1.2, the source of voiced speech refers to the glottis, whereby the source signal is generated by the vocal fold vibration. In this thesis, the source signal is not separated from the acoustic speech signal. Instead, the speech signal including all information is used directly for feature extraction. Four different voice quality feature types are extracted for this thesis. Two of them are the most common descriptors that characterise the voice, namely *jitter* and *shimmer* (Section 4.4.1). Both parameters are related to the period-to-period variation. Further, additive noise in the voice signal is measured by the *harmonics-to-noise ratio* (Section 4.4.2). Finally, the fourth feature type of this group is referred to as the *cepstral peak prominence* (Section 4.4.3), a parameter which is based on the cepstrum.

Voice quality feature extraction is performed using the analysis tool Praat (Boersma, 2002). Since these types of features refer to the source signal, which is periodic, there should be enough consecutive periods per analysis window to obtain accurate measures. In the case of jitter, shimmer, and cepstral peak prominence, the analysis window length is set to 50 ms with a frame-shift of 10 ms. For the harmonics-to-noise ratio, the optimal standard setting of Praat is used with 4.5 periods per window and also a frame-shift of 10 ms. The following sections give details on feature computation.

4.4.1 Jitter and Shimmer

Jitter and *shimmer* are the most common descriptors that characterise the voice quality. The computation of both parameters relates to variations regarding the oscillation of the vocal folds (Lass, 2014, p. 301).

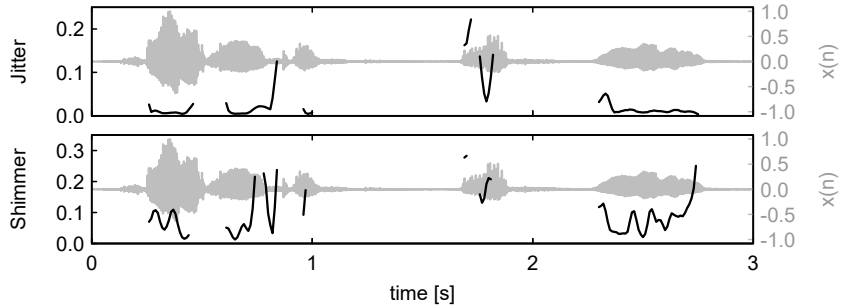


Figure 4.11 Waveform (background), jitter, and shimmer (foreground) of a speech signal

Jitter is defined as the period-to-period variation in vocal fold frequency. The local jitter of a frame can be computed by the average absolute difference between consecutive periods normalised by the average period length (Boersma, 2002), expressed as:

$$jitter = \frac{\frac{1}{N-1}\sum_{n=2}^{N}|T_0(n) - T_0(n-1)|}{\frac{1}{N}\sum_{n=1}^{N}T_0(n)}, \tag{4.13}$$

where T_0 is the duration of the nth period and N represents the number of periods (Lass, 2014, p. 302).

Similarly, shimmer is a measure related to the period-to-period variation in the amplitude of a voice. The local shimmer of a frame is obtained by the average absolute difference between the amplitudes of consecutive periods, divided by the average amplitude (Boersma, 2002). The definition can be written as:

$$shimmer = \frac{\frac{1}{N-1}\sum_{n=2}^{N}|A(n) - A(n-1)|}{\frac{1}{N}\sum_{n=1}^{N}A(n)}, \tag{4.14}$$

where $A(n)$ is the amplitude of the nth period and N represents the number of periods.

Figure 4.11 illustrates the local jitter and shimmer for a speech signal. Excessive amounts of both parameters are generally considered to be indicative of rough or hoarse voice quality (Lass, 2014, p. 300). It is assumed that irregularities in frequency and amplitude are consequences of biomechanical (vocal fold asymmetry), neurogenic (involuntary activities of muscles), and aerodynamic (fluctuations of airflow) factors (Honda, 2008, p. 11). Moreover, Orlikoff and Baken (1989) showed that heart rate can have an influence on the jitter.

In fact, findings concerning the effects of cognitive load on jitter and shimmer are not consistent across studies (cf. Section 2.3.2.3). For the speech material used

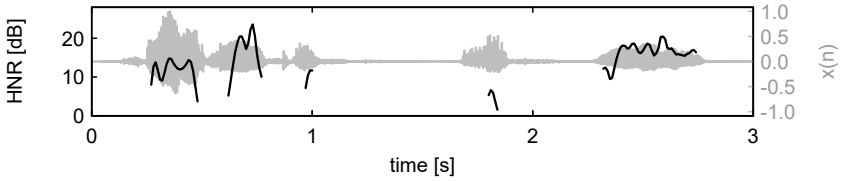


Figure 4.12 Waveform (background) and harmonics-to-noise ratio (foreground) of a speech signal

in this thesis, a decreasing trend can be observed for both parameters from low to medium cognitive load (cf. Figure A.2 in Appendix). Interestingly, the influence of high cognitive load conditions caused by time pressure (CLSE-Time) leads in turn to a significant increase in jitter and shimmer. A monotonically decreasing trend across all cognitive load levels can only be observed for shimmer on CLSE-Dual, jitter on CLSE-Span, and both on CoLoSS. In these cases, differences between medium and high cognitive load are not significant, but the results generally indicate that speech includes less rough or hoarse characteristics as cognitive load increases.

4.4.2 Harmonics-to-Noise Ratio

The *harmonics-to-noise ratio* (HNR) is a measure of additive noise in the voice signal. The logarithmic HNR is computed for this thesis by using the autocorrelation function approach as described in Boersma (1993). In more detail, autocorrelation corresponds to the power of the signal at lag $\tau = 0$. If the period T_0 of the signal is uncorrelated with added noise, the autocorrelation of the resulting signal equals the sum of the autocorrelations of its parts. That is, the autocorrelation at the fundamental period T_0 refers to the power of the periodic component (i.e., harmonic) of the signal, whereas its complement represents the power of the noise component. The logarithmic HNR in dB can be written as (Boersma, 1993):

$$HNR = 10 \cdot \log_{10} \left(\frac{ACF(T_0)}{ACF(0) - ACF(T_0)} \right). \tag{4.15}$$

As a consequence, an equal proportion of the energy in the harmonics and in the noise will result in HNR of 0 dB. Figure 4.12 illustrates the harmonics-to-noise ratio for a speech signal. This parameter can be considered to be an acoustic correlate for breathiness and roughness (Abhang et al., 2016, p. 65).

Means and 95% confidence intervals of HNR in terms of different cognitive load levels are found in Appendix (Figure A.2). As cognitive load increases, no reliable trend can be observed for the data of CLSE-Time: there is an increase from low to medium and a decrease from medium to high cognitive load, whereby differences between the levels are statistically significant. For the remaining corpora, the mean HNR shows a monotonically increasing trend as cognitive load increases. In this context, significant differences can only be observed for CoLoSS (from low to medium and low to high cognitive load). The results suggest a less breathy voice quality as cognitive load increases.

4.4.3 Cepstral Peak Prominence

The *cepstral peak prominence* (CPP), introduced by Hillenbrand et al. (1994), compares the amplitude of the cepstral peak with the cepstrum background. As described in Section 4.1.2, the cepstrum $c(n)$ is computed by taking the inverse DFT of $\log|X(k)|$, where $\log|X(k)|$ is the logarithm of the magnitude response obtained by taking the DFT of a signal. The cepstrum contains a large peak located at the quefrency q_0, which corresponds to the fundamental frequency F_0. The cepstrum background is represented by a fitted regression line through the cepstrum. By taking the cepstral peak $c(q_0)$ and the fitted regression line $\hat{c}(n)$ into account, the CPP is obtained by:

$$CPP = c(q_0) - \hat{c}(q_0), \tag{4.16}$$

where $\hat{c}(q_0)$ represents the corresponding value on the regression line, i.e., the predicted magnitude for the quefrency at the cepstral peak.

Figure 4.13 illustrates the cepstral peak prominence for a speech signal. Note, although this parameter can be computed for the overall signal, only voiced frames are considered due to the focus on voice quality. It has been shown that CPP is highly correlated with breathiness (Hillenbrand and Houde, 1996) and the degree of dysphonia (Heman-Ackah et al., 2002). Moreover, the authors Yap et al. (2011b) demonstrated that CPP is a promising parameter for automatic cognitive load classification.

An increasing trend of the CPP can be observed with regard to the level of cognitive load (cf. Figure A.2 in Appendix). However, for both CLSE-Time and CLSE-Dual, there is an insignificant drop from medium to high cognitive load. For all speech corpora used in this thesis, CPP values exhibit statistically significant differences between low and high cognitive load conditions. A monotonically increasing trend across all cognitive load levels can only be observed for CLSE-Span (not significant between low and medium load) and CoLoSS (not significant between medium and high load). The results indicate that speech tends to involve less breathy characteristics as cognitive load increases.

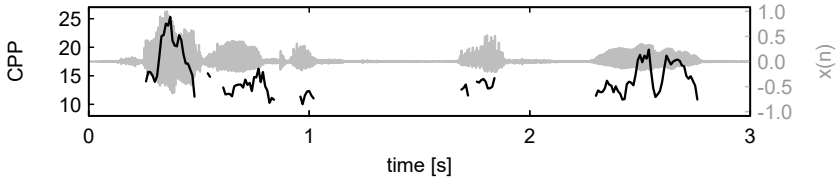


Figure 4.13 Waveform (background) and cepstral peak prominence (foreground) of a speech signal

4.5 Teager Energy Operator Based Features

The process of human speech production can be explained by the source-filter model (Section 2.3.1.2). The theory behind this traditional model assumes that the source of speech production refers to the motion of the vocal folds and the constriction of the vocal tract.

However, there is also a different view in this regard. According to studies by Teager and Teager (1990), human speech production is related to airflow patterns in the vocal tract. It was suggested that the true source of sound refers to the interactions between vortex-flows, which are non-linear. Based on these preliminary considerations, the Teager energy operator (TEO) has been developed to measure the instantaneous energy of such non-linear interactions. For a discrete-time signal $x(n)$, the TEO is expressed by the following equation (Kaiser, 1993):

$$\Psi[x(n)] = x^2(n) - x(n+1)x(n-1). \tag{4.17}$$

Since the TEO is computed per sample, the result has a very high resolution. The operator is usually applied to a bandpass filtered speech signal in order to determine the energy of the non-linear flow for resonant frequencies. Non-uniform filterbanks are commonly used, which are justified from a human auditory perception point of view.

One can assume that changes in the human vocal system physiology induced by stress will affect the airflow patterns in the vocal tract (Zhou et al., 2001). As a consequence, TEO-based features have gained interest for the automatic classification of stress (Zhou et al., 1998a,b; Fernandez and Picard, 2003; Ruzanski et al., 2005). Moreover, some works demonstrated the utilisation of the TEO for clinical depression detection (e.g., Low et al., 2009), emotion recognition (e.g., Nwe et al., 2003), robust speech recognition (e.g., Jabloun and Cetin, 1999; Dimitriadis et al., 2005), and the classification of pathological and normal voices (e.g., Salhi and Cherif, 2013).

To the best knowledge of the author of this thesis, TEO-based features have not yet been used for cognitive load recognition. In this thesis, such features are

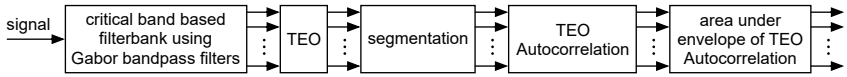


Figure 4.14 TEO-CB-Auto-Env feature extraction. Adapted from Zhou et al. (2001)

investigated for the first time in this field. The following three TEO-based features are extracted: *critical band based TEO autocorrelation envelope area* (Section 4.5.1) and *nonlinear time domain LFPC* as well as *nonlinear frequency domain LFPC* (Section 4.5.2). All three feature types involve bandpass filtering, which is achieved by short-time spectral analysis. For this purpose, a Hamming window function is chosen with a window length of 25 ms and a frame-shift of 10 ms. Since there is no toolkit available that provides the extraction of these features, they have been implemented for this thesis using the free software environment R (R Core Team, 2015).

4.5.1 Critical Band Based TEO Autocorrelation Envelope Area

Zhou et al. (2001) introduced different TEO-based features with the goal to investigate variations of the airflow energy within the vocal tract for speech under stress. In this context, one of the most promising features in terms of accuracy and reliability is known as the *critical band based TEO autocorrelation envelope area* (TEO-CB-Auto-Env). The process chain of TEO-CB-Auto-Env includes five steps as depicted in Figure 4.14.

In the first step, the so-called critical band based filterbank is applied in order to filter specific frequency ranges of a voiced speech signal. The concept of critical bands is associated with the assumption that the human auditory system performs a filtering operation on the entire audible frequency range; the design of such a filterbank involves empirical research. The centre frequency and bandwidth of each band are given in Table 4.2. Each of the 16 filters is a Gabor bandpass filter. This type of filter is characterised by a Gaussian shape, i.e., the band fades out at the sides. In discrete form, the Gabor filter can be expressed as $h(n) = \exp\left(-2\pi(b\,Tn)^2\right)\cos\left(2\pi f_c\,Tn\right)$, with $-N \leq n \leq N$, where T is the sampling period, f_c is the centre frequency, and b is the bandwidth of the filter (Maragos et al., 1993).

In the second step of the TEO-CB-Auto-Env process chain, the TEO is applied to each filter output; this yields 16 different TEO streams. These streams are then segmented into frames with a fixed length. A frame length of 25 ms and a frame-shift of 10 ms is chosen. Next, for each frame of the TEO streams, the normalised

Table 4.2 Critical band based filterbank. Adapted from Zhou et al. (2001)

	Critical band frequency (Hz)	
Band	Centre	Bandwidth
1	150	100
2	250	100
3	350	100
4	450	110
5	570	120
6	700	140
7	840	150
8	1000	160
9	1170	190
10	1370	210
11	1600	240
12	1850	280
13	2150	320
14	2500	380
15	2900	450
16	3400	550

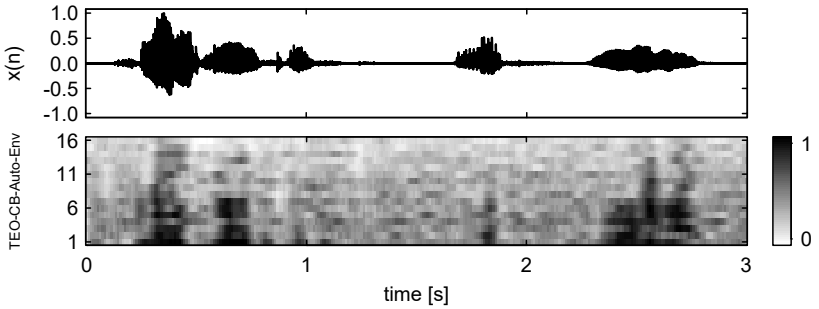


Figure 4.15 Waveform and TEO-CB-Auto-Env heat map of a speech signal

autocorrelation function is applied. By computing the area under the envelope of the normalised autocorrelation function, 16 TEO-CB-Auto-Env contours are obtained. Figure 4.15 illustrates the 16 TEO-CB-Auto-Env parameters for a speech signal in the form of a heat map. This type of feature is actually extracted from voiced speech. However, the figure should give an impression of how the feature behaves on the entire signal.

Statistical analyses of TEO-CB-Auto-Env with respect to cognitive load are found in Appendix (Figure A.6). The 16 parameters show almost no relationship to cognitive load for the data of CLSE-Span and CoLoSS. It can be assumed that the slightly higher correlation values obtained on CLSE-Time and CLSE-Dual are linked to the suitability of TEO features for tasks in which individuals have to perform under stressful conditions.

4.5.2 Nonlinear Log-Frequency Power Coefficients

The detection of stress and emotion in speech has been investigated by Nwe et al. (2003) using the non-linear properties of the Teager energy operator (TEO) in combination with *Log-Frequency Power Coefficients* (LFPCs). LFPCs simulate the logarithmic filtering characteristics of the human auditory system. This is achieved by accumulating the frequency spectrum of a discrete-time signal into a bank of log-frequency filters.

For LFPC implementation, a log-frequency filterbank has been designed by the author of this thesis. A set of Q bandpass filters can be generated as follows (Rabiner and Juang, 1993, p. 78):

$$b_1 = C, \tag{4.18a}$$

$$b_i = \alpha b_{i-1}, \quad 2 \leq i \leq Q, \tag{4.18b}$$

$$f_i = f_1 + \sum_{j=1}^{i-1} b_j + \frac{(b_i - b_1)}{2}, \tag{4.19}$$

where C and f_1 are the predefined bandwidth and the first centre frequency parameters, respectively, and α is the predefined logarithmic growth factor. The filterbank was specified by $C = 54$ Hz, $f_1 = 127$ Hz, $\alpha = 1.4$, and $Q = 12$. Table 4.3 gives an overview of the resulting log-frequency filterbank. The parallel set of bandpass filters covers the spectrum from 100 Hz to about half of the sampling frequency (i.e., Nyquist frequency).

The energy concerning a particular filter i in the filterbank is given by (Nwe et al., 2003):

$$E(i) = \sum_{k=f_i-\frac{b_i}{2}}^{f_i+\frac{b_i}{2}} X(k)^2, \quad i = 1, 2, \ldots, 12, \tag{4.20}$$

where $X(k)$ is the kth spectral component; f_i and b_i correspond to the ith centre frequency and bandwidth, respectively. In order to obtain 12 LFPCs for each

Table 4.3 Log-frequency filterbank

	Log-frequency (Hz)	
Band	Centre	Bandwidth
1	127	54
2	192	76
3	283	106
4	410	148
5	587	207
6	836	290
7	1185	407
8	1673	569
9	2356	797
10	3312	1116
11	4651	1562
12	6525	2187

frame, the logarithmic power is calculated as follows (Nwe et al., 2003):

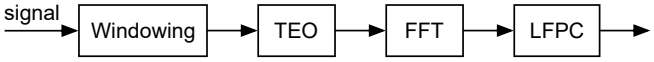
$$P(i) = 10 \cdot \log_{10}\left(\frac{E(i)}{N_i}\right), \quad i = 1, 2, \ldots, 12, \tag{4.21}$$

where N is the number of spectral components within the boundaries of the ith filter.

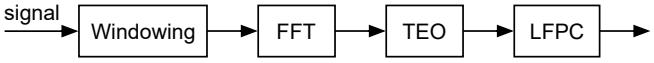
As aforementioned, the non-linear properties of the TEO can be combined with LFPCs. Although the TEO is typically applied in the time domain, Nwe et al. (2003) investigated the operator in both the time domain and the frequency domain. The corresponding feature types are referred to as *nonlinear time domain LFPC* (NTD-LFPC) and *nonlinear frequency domain LFPC* (NFD-LFPC). The process chains are found in Figure 4.16. NTD-LFPC implies that the TEO is applied to a windowed signal in the time domain and, afterwards, FFT is performed to compute the LFPCs. In contrast, NFD-LFPC requires that the windowed signal is first transformed to the frequency domain by FFT and the TEO is then applied to the frequency spectrum; at the end, LFPCs are computed.

Both feature types result in 12 low-level parameters according to the number of bandpass filters. Figure 4.17 illustrates the waveform of a speech signal and the corresponding nonlinear LFPC contours in the form of a heat map.

Correlation measurements between nonlinear LFPCs and cognitive load indicate the importance of TEO-based features for tasks where individuals have to cope with stressful conditions, which are reflected by variants of the Stroop test (cf. Figure A.7 and A.8 in Appendix). More precisely, in the case of CLSE-Time,



(a) Nonlinear time domain LFPC



(b) Nonlinear frequency domain LFPC

Figure 4.16 Nonlinear LFPC feature extraction

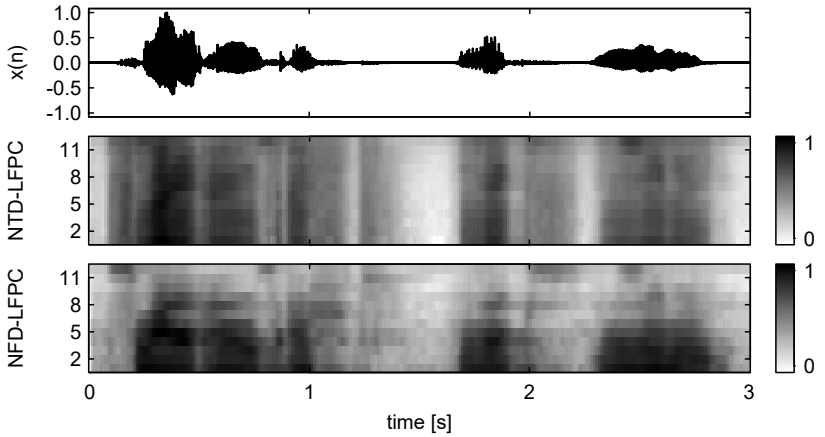


Figure 4.17 Waveform, NTD-LFPC heat map, and NFD-LFPC heat map of a speech signal

a moderate (positive) up to a high correlation is obtained by almost all mean NTD-LFPCs as well as mean NFD-LFPCs. For CLSE-Dual, mean NTD-LFPCs and mean NFD-LFPCs exhibit a low (positive), with a tendency to moderate, relationship. For both Stroop test datasets, however, statistically significant differences are solely observed between low and medium as well as low and high cognitive load ($p < 0.05$)—with only a few exceptions. This means that adding time pressure or a secondary task to conditions where the word meaning and font colour are incongruent (medium load) has almost no effect on nonlinear LFPCs. Apart from that, CLSE-Span and CoLoSS yield negligible correlation values for both feature types. For all tasks, only a little if any relationship to cognitive load can be observed for variations in nonlinear LFPCs.

4.6 Derived Features

The low-level features introduced in the previous sections are represented by separate frames, where each frame is assigned to a particular index on the timeline. At this point, there is no information about the relationship between past and future frames. In order to obtain this contextual information, delta coefficients are derived from the time series as described in Section 4.6.1.

Another issue refers to the representation of different feature types. Audio analysis systems usually require data to be in the form of constant-dimensional feature vectors. Moreover, low-level feature contours may contain redundant information, which does not contribute to relevant patterns in the data. Feature summarisation by applying so-called 'functionals' is a popular approach in this regard (Section 4.6.2).

4.6.1 Time Derivatives

In addition to the basic low-level feature types, information about their temporal evolution can be determined to describe the dynamic characteristics. In this thesis, this is achieved by computing the so-called delta coefficients using the following regression formula (Young et al., 2006, p. 68):

$$\Delta(n) = \frac{\sum_{i=1}^{W} i \left(x(n+i) - x(n-i)\right)}{2\sum_{i=1}^{W} i^2}, \tag{4.22}$$

where $\Delta(n)$ is a delta coefficient at frame n computed in terms of the corresponding frames $x(n-W)$ to $x(n+W)$. As a consequence, the equation relies on a number of past and future frames depending on the window size W. The same formula can be applied to the first order derivative $\Delta(n)$ to obtain the second order derivative $\Delta\Delta(n)$, also known as the acceleration coefficients.

Based on Equation (4.22), delta regression has been implemented for this thesis. Delta and acceleration coefficients are computed for each low-level feature; the chosen window size is $W = 2$.

4.6.2 Statistical Functionals

There are various approaches to process features for use in machine learning. Some systems consider each low-level feature vector separately and operate directly on frame-level. Thus, frame-wise results are combined into a single result using late fusion schemes, or dynamic approaches are used with temporal alignment provided by methods such as Hidden Markov Models.

Table 4.4 Overview of statistical functionals used

Functional group	Functional type
Moments	Mean, standard deviation, skewness, kurtosis
Extrema	Minimum, maximum, range, maximum-mean difference, mean-minimum difference, relative position of global minimum and maximum
Percentiles	25th, 50th, and 75th percentile
Regression	Simple linear regression slope and intercept

Another method is to combine each low-level feature vector into a single feature vector, which is then used for training or testing. This technique is also referred to as 'supra-segmental' analysis (Schuller, 2013, p. 19). There are several ways to combine feature vectors. For instance, they can be concatenated to a higher-dimensional vector. This is only feasible if the length of audio segments is fixed because learning algorithms usually require constant-dimensional feature vectors. Due to the nature of speech, however, the length of audio segments vary considerably. To realise a constant dimensionality for feature vectors, it is common practice to summarise the low-level feature vectors by applying *statistical functionals* to them. Typical functionals are the minimum, maximum, and the mean.

In this thesis, the concept of statistical functionals is followed. Table 4.4 gives an overview of those functionals that are used for this thesis; four different functional groups are considered: moments, extrema, percentiles, and regression.[2] In the following sections, a brief introduction to the mathematics behind these statistical descriptors is given.

4.6.2.1 Moments

The arithmetic mean is referred to as the first moment. It is given by:

$$\mu = \frac{1}{N} \sum_{n=0}^{N-1} x(n). \tag{4.23}$$

The spread of the feature around the arithmetic mean is provided by the variance and the standard deviation. The variance, also known as the second moment, is

[2]The Java statistics package `org.apache.commons.math3.stat` is used for computation.

defined by:

$$\sigma^2 = \frac{1}{N} \sum_{n=0}^{N-1} (x(n) - \mu)^2. \tag{4.24}$$

The standard deviation σ can then be calculated by taking the square root of the variance σ^2. The third standardised moment is the skewness—a measure of the asymmetry of the data distribution. It is defined by:

$$skewness = \frac{1}{N} \sum_{n=0}^{N-1} \left(\frac{x(n) - \mu}{\sigma} \right)^3. \tag{4.25}$$

The kurtosis is referred to as the fourth standardised moment. It measures the peakedness or flatness relative to the normal distribution. The definition is:

$$kurtosis = \frac{1}{N} \sum_{n=0}^{N-1} \left(\frac{x(n) - \mu}{\sigma} \right)^4. \tag{4.26}$$

4.6.2.2 Extrema

The global minimum min_x and maximum max_x of a given set of data points are the smallest and the largest value, respectively. Both extrema can be used for the computation of the range ($max_x - min_x$), maximum-mean difference ($max_x - \mu_x$), and mean-minimum difference ($\mu_x - min_x$).

In addition, data can be described using information on the position of extrema. The relative position of min_x and max_x is obtained by the position index of the corresponding extremum divided by the maximum number of indices.

4.6.2.3 Percentiles

Percentiles reflect the location of an observation in a distribution. A percentile is a value for which a given percentage of the data is less than or equal to. For percentile determination, the values are organised in ascending order. The percentile location can be calculated by $i = (P \cdot N)/100$, where P is the percentile of interest and N is the total number of values (Black, 2012, p. 57). If the percentile location results in a whole number, the Pth percentile is the average of the value at location i and location $i + 1$. In the case where the location i is not a whole number, the floor of $i + 1$ can then be used.

The 25th percentile, also known as the first quartile, separates the first quarter of the data from the upper three-fourths. The 50th percentile is the second quartile, also known as the median. The 75th percentile is the third quartile; it divides the first three-quarters of the data from the last quarter.

4.6.2.4 Regression

Linear regression provides useful information about the global progression of a feature contour. The simple linear regression model is given by (Montgomery et al., 2012, p. 12):

$$y = \beta_0 + \beta_1 x + \epsilon, \tag{4.27}$$

where y is the dependent variable, β_0 is the y-intercept, β_1 is the slope of the regression line, x is an independent variable (regressor), and ϵ represents a random error component.

In order to estimate β_0 and β_1, the method of least squares can be used. In this connection, the goal is to estimate β_0 and β_1 in the way that the sum S of the squared distance between the observations y_i and the predicted response of the model is a minimum. The least squares criterion for N pairs of data is expressed as (Montgomery et al., 2012, p. 13):

$$S(\beta_0, \beta_1) = \sum_{i=1}^{N} (y_i - \beta_0 - \beta_1 x_i)^2. \tag{4.28}$$

The motivation behind this method is to determine parameters by choosing the regression line that is closest to all data points.

4.7 Process Chain for Feature Extraction

The previous sections of this chapter introduced the fundamentals of audio signal processing (Section 4.1) and described those feature types that are investigated in this thesis (Section 4.2 to 4.5). Moreover, feature post-processing has been taken into consideration (Section 4.6). In this section, the chain of feature extraction is presented, which has been implemented for this thesis. The process chain consists of seven components as depicted in Figure 4.18. Features are extracted in two ways: *Low-level feature extraction* and *ASR-based feature extraction*.

Low-level feature extraction results in a time series $x(n)$ represented by a sequence of frames. In order to eliminate artefacts in the data, the low-level feature is first smoothed by applying the *SMA* filter (Section 4.1.4). Next, *Delta* derives delta coefficients (Section 4.6.1) from a sequence of frames—the output is a time series, where each index refers to the relationship between past and future frames of the input data. Applying this component multiple times leads to higher-order deltas. To obtain relevant segments from the speech signal, voice activity detection (*VAD*, Section 4.1.5) is applied to both the smoothed low-level feature and the corresponding delta features. In this thesis, only voiced speech is considered based on the segmental boundaries of the fundamental frequency

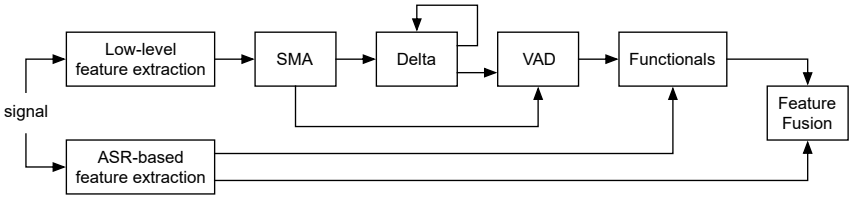


Figure 4.18 Process chain for feature extraction

(Section 4.2.2). The aim of this restriction is attributed to the assumption that unvoiced speech frames result in unsteady characteristics for almost all low-level features (cf. Section 2.3.1.2—noise component of the source-filter model); in particular in the case of formants (Section 4.3.3) and voice quality features (Section 4.4), voice information is required for a robust estimate. In addition, the *VAD* component is applied for the extraction of different speech events (Section 4.2.3.2). Finally, the processed low-level features are transferred to the *Functionals* component (Section 4.6.2), which summarises the sequence of data points (or frames) by applying a number of statistical functions to them.

In the case of *ASR-based feature extraction* (Section 4.2.3.1), features are computed on the basis of phonemes. Duration types and speaking rate contours are directly passed to the *Functionals* component. The frame sequence of speaking rate contours is not processed by *SMA*, *Delta*, or *VAD* for the following reasons: (1) *VAD* is part of the ASR system anyway; (2) *SMA* and *Delta* restrict the number of observations at the beginning and the end of the frame sequence due to the window analysis—since the number of frames of speaking rate contours is comparatively low, *SMA* would cause a loss of important information and *Delta* would not produce reliable information. Further, static ASR-based features (e.g., silent pause frequency) are extracted. Since these parameters are static and describe the speech signal with a single value, post-processing is not required.

Once feature extraction is completed, all resulting feature vectors are concatenated by the *Feature Fusion* component. The result is a single feature vector, where the dimensionality corresponds to the number of all features.

Chapter 5

Feature Sets and Relevance Analysis

A central aim of this thesis is the evaluation of speech features for the automatic recognition of cognitive load. In this regard, the static modelling approach is followed (Section 6.1.4), which generally requires constant-dimensional feature vectors. As pointed out in the previous chapter, constant dimensionality is realised by the concept of statistical functionals (Section 4.6.2).

This chapter starts with an introduction of three hand-crafted feature sets designed for cognitive load (CL) recognition. While CL-Extended (Section 5.1.1) represents a generic feature set including the amount of all features that are investigated in this thesis, the feature subsets CL-Base (Section 5.1.2) and CL-Stress (Section 5.1.3) address solely the aspects of cognitive load and stress, respectively. Further, feature relevance analysis is conducted to give an insight into the importance of features for different cognitive load tasks. This includes correlation measurements (Section 5.2) reflecting the strength of the relationship between features and the level of cognitive load. In addition, a broader context of feature relevance is considered by analysing feature groups and feature types using an entropy-based approach (Section 5.3).

5.1 Feature Set Composition

It can be assumed that a context-dependent combination of speech features yields a system improvement in terms of the automatic cognitive load recognition. Therefore, three feature sets are defined (Section 5.1.1 to 5.1.3) and evaluated in the experiments (Chapter 6). The composition of the proposed sets is hand-crafted based on findings from the scientific literature.

5.1.1 CL-Extended

The CL-Extended feature set contains the amount of all features that are extracted for this thesis. Apart from features that are assumed to be indicative of cognitive

Table 5.1 CL-Extended feature set

Feature group	Feature type	Functionals	# Features
Low-level			
Prosodic	F_0+Δ+ΔΔ	A	48
Prosodic	Intensity+Δ+ΔΔ	A	48
Spectral	Spectral centroid+Δ+ΔΔ	A	48
Spectral	12 MFCC+Δ+ΔΔ	A	576
Spectral	Formants $F_{1-3}(f,b)$+Δ+ΔΔ	A	288
Voice quality	Jitter+Δ+ΔΔ and Shimmer+Δ+ΔΔ	A	96
Voice quality	HNR+Δ+ΔΔ	A	48
Voice quality	CPP+Δ+ΔΔ	A	48
TEO	16 TEO-CB-Auto-Env+Δ+ΔΔ	A	768
TEO	12 NTD-LFPC+Δ+ΔΔ	A	576
TEO	12 NFD-LFPC+Δ+ΔΔ	A	576
High-level			
Prosodic	4 Duration types	B	40
Prosodic	2 Speaking rates	C	18
Prosodic	3 Segment lengths	D	12
Prosodic	11 Static features		11
$\sum$			3,201

Information on functional sets are found in Table 5.2

load (cf. Section 2.3.2), this set also includes paralinguistically motivated parameters from the field of automatic stress detection, namely Teager energy operator (TEO) based features (cf. Section 4.5). Moreover, CL-Extended forms the basis for evaluating various feature subsets, such as other hand-crafted feature sets, feature groups, feature types, or subsets obtained by applying feature selection algorithms.

The CL-Extended feature set is summarised in Table 5.1. Each feature type refers to a particular feature group: prosodic features (Section 4.2), spectral features (Section 4.3), voice quality features (Section 4.4), and TEO-based features (Section 4.5). Further, a distinction was made between low-level and high-level features. Within the scope of this thesis, low-level features refer to the acoustic signal directly, whereas high-level features refer to different representations of speech, such as phonemes, syllables, or voiced segments; here, all duration-based features (Section 4.2.3) are classified as high-level. For all low-level features, the first order derivative (Δ) and second order derivative (ΔΔ) are added. In the case of windowed high-level features (speaking rates), it was decided to omit this contextual information due to the low number of frames. In other words:

Table 5.2 Overview on functional sets

	Functional set			
Functional	A	B	C	D
Mean	✓	✓		✓
Standard deviation	✓	✓	✓	✓
Skewness	✓			
Kurtosis	✓			
Minimum	✓	✓	✓	✓
Maximum	✓	✓	✓	✓
Range	✓			
Maximum-mean difference	✓	✓	✓	
Mean-minimum difference	✓	✓	✓	
Relative position of minimum	✓	✓	✓	
Relative position of maximum	✓	✓	✓	
25th percentile	✓			
50th percentile	✓			
75th percentile	✓			
Simple linear regression slope	✓	✓	✓	
Simple linear regression intercept	✓	✓	✓	

time derivatives are associated with discarding the first and the last frame (cf. Equation (4.22)), which, in turn, may result in too few data points for computing informative statistical functionals.

Different sets of functionals are defined in Table 5.2. Functional set A applies all 16 functionals to the low-level features with the goal to obtain as much information as possible from the time series. The functionals contained in B, C, and D constitute functional subsets that are applied only to the high-level features. In these cases, some functions concerning the data distribution are omitted, because reliable information can hardly be derived due to the comparatively small number of observations, especially when the utterance duration is very short. The lowest number of observations is generally given by segment lengths—only four functionals are included in set D. Functional set C is applied to the two types of speaking rates (phoneme-based and syllable-based). In contrast to set B, the first moment (mean) in set C is omitted since the average speaking rate is already included in the static features. In this connection, a static feature gives a single numerical value per utterance instead of a series of observations. With reference to Section 4.2.3, the following features are classified as static: utterance duration, 2 speaking rates (phoneme-based and syllable-based), 2 articulation rates (phoneme-based and syllable-based), 5 silent pause features, and the onset latency.

In summary, the CL-Extended feature set contains 65 low-level features and 65 corresponding delta and acceleration coefficients. 16 functionals are applied to these 195 contours, which gives a total of 3,120 features. Further, the set contains 81 high-level features: 4 duration types (phonemes, consonants, vowels, and syllables) for which 10 functionals are applied; 2 speaking rates (phonemes and syllables) with 9 functionals; 3 segment length features (speech, voiced, and unvoiced) with 4 functionals; 11 static features (10 phoneme-level features and the onset latency). The CL-Extended feature set yields a total of 3,201 features.

5.1.2 CL-Base

The CL-Base feature set represents a subset of CL-Extended (Section 5.1.1). It contains only those features that are assumed to be indicative of cognitive load. The design is based on studies from the scientific literature concerning the influence of cognitive load on speech (Section 2.3.2). Note, this does not necessarily rule out the suitability of CL-Base for other scenarios.

Since the goal is to compose a reduced set of features, only the first order derivative (Δ) is added. Further, the centre frequency and bandwidth of the third formant are excluded, because previous studies reported that there are no significant effects of cognitive load on these parameters. The number of voice quality features are reduced as well: harmonics-to-noise ratio (HNR) and cepstral

Table 5.3 CL-Base feature set

Feature group	Feature type	Functionals	# Features
Low-level			
Prosodic	F_0+Δ	A	32
Prosodic	Intensity+Δ	A	32
Spectral	Spectral centroid+Δ	A	32
Spectral	12 MFCC+Δ	A	384
Spectral	Formants $F_{1-2}(f,b)$+Δ	A	128
Voice quality	Jitter+Δ and Shimmer+Δ	A	64
Voice quality	CPP+Δ	A	32
High-level			
Prosodic	4 Duration types	B	40
Prosodic	2 Speaking rates	C	18
Prosodic	3 Segment lengths	D	12
Prosodic	11 Static features		11
$\sum$			785

Information on functional sets are found in Table 5.2

peak prominence (CPP) are measures of energy noise in the signal and both are (negatively) correlated with breathiness; since HNR and CPP are highly correlated (cf. Samlan et al., 2013) and CPP seems to be very robust against changes in utterances (cf. Fraile and Godino-Llorente, 2014), it was decided to exclude HNR from the feature set. So far, there is no evidence regarding the suitability of Teager energy operator based features (TEO) for cognitive load recognition. Consequently, the TEO feature group is omitted.

The CL-Base feature set is summarised in Table 5.3. It contains 22 low-level features and 22 corresponding delta coefficients. For each of these 44 contours, 16 functionals are applied resulting in 704 features. All high-level features of CL-Extended are also included in CL-Base since consistent effects of cognitive load on these types of features have been demonstrated in the past (cf. Section 2.3.2.2). In total, the CL-Base feature set contains 785 features.

5.1.3 CL-Stress

The design of the CL-Stress feature set is motivated by the hypothesis that a subset of CL-Extended (Section 5.1.1) is more suitable for use in stress-related environments. It has been found that features based on the Teager energy operator (TEO) are effective for stress detection (cf. Section 4.5). Consequently, TEO-based features are included in CL-Stress—and that is the major difference by

Table 5.4 CL-Stress feature set

Feature group	Feature type	Functionals	# Features
Low-level			
Prosodic	F_0+Δ	A	32
Prosodic	Intensity+Δ	A	32
Spectral	12 MFCC+Δ	A	384
Spectral	Formants $F_{1-2}(f,b)$+Δ	A	128
Voice quality	Jitter+Δ and Shimmer+Δ	A	64
TEO	16 TEO-CB-Auto-Env+Δ	A	512
TEO	12 NTD-LFPC+Δ	A	384
High-level			
Prosodic	4 Duration types	B	40
Prosodic	2 Speaking rates	C	18
Prosodic	3 Segment lengths	D	12
Prosodic	11 Static features		11
$\sum$			1,617

Information on functional sets are found in Table 5.2

comparison with CL-Base (Section 5.1.2). Since the TEO is typically applied in the time domain, the rather experimental feature NFD-LFPC, which applies the TEO in the frequency domain, is omitted. Compared to CL-Extended and CL-Base, this feature set contains fewer spectral features. Moreover, only the two most common voice quality features are included, namely jitter and shimmer.

Table 5.4 summarises the CL-Stress feature set. It contains 48 low-level features and 48 corresponding delta coefficients. By applying 16 functionals to the 96 contours, 1,536 features are obtained. As in CL-Extended and CL-Base, all 81 high-level features are included. In sum, the CL-Stress feature set contains 1,617 features.

5.2 Monotonic Relationship

The most effective features out of all extracted features (Section 5.1.1) are now addressed, whereby the relationship between features and the level of cognitive load is of interest. To this end, the datasets presented in Chapter 3 are used. In order to remove the inter-speaker variability and environmental mismatch, all features are speaker-normalised for each cognitive load task individually using z-score transformation ($\mu = 0, \sigma = 1$). Regarding the CLSE database, only the data of the training and development set can be used, because speaker information is not included in the test partition (cf. Section 3.1.2). It is to note that in the case of the CoLoSS corpus only categorical labels are considered (Section 3.2.5).

Due to the nature of ordinal cognitive load labels, the absolute Spearman's correlation coefficient is chosen as the relevance criterion. The Spearman's correlation coefficient CC_S determines the strength and direction of the monotonic relationship. Accordingly, its numerical value ranges from -1 to 1. A value near zero indicates that there is no relationship between the feature and the level of cognitive load. For CC_S determination, let $x(n)$ and $y(n)$ be the feature and the level of cognitive load, respectively. The coefficient CC_S can be computed by first replacing $x(n)$ and $y(n)$ with their ranks and then applying Pearson's equation (Pearson, 1895):

$$CC_P = \frac{\sum_{n=0}^{N-1} \big(x(n) - \mu_x\big)\big(y(n) - \mu_y\big)}{\sqrt{\sum_{n=0}^{N-1} \big(x(n) - \mu_x\big)^2} \sqrt{\sum_{n=0}^{N-1} \big(y(n) - \mu_y\big)^2}}. \tag{5.1}$$

In the following sections, feature relevance analysis is carried out—the ten most important features are determined using R (R Core Team, 2015). The features are ranked according to the absolute CC_S within the context of a particular task and, in addition, across all tasks. Since each task is represented by a different corpus, the analyses are referred to as 'within-corpus feature ranking'

(Section 5.2.1) and 'cross-corpus feature ranking' (Section 5.2.2). Significance of correlation was tested (t-test) with the null hypothesis that there is no monotonic relationship between the feature and the level of cognitive load; information on p-values are given in the corresponding tables.

5.2.1 Within-Corpus Feature Ranking

This analysis aims at identifying the most relevant features for each single task (Table 5.5 to 5.8). Correlation measurements are reported for the top ten features on a given task along with presenting their correlation on the remaining tasks.

Regarding the data of CLSE-Time (Table 5.5), the ten most relevant features are quite obviously nonlinear LFPCs. These features exhibit a high positive relationship with the level of cognitive load. The strongest correlation is obtained by the 25th percentile of the NFD-LFPC 11. However, all ten features show low correlation values for CLSE-Dual, though still statistically significant. In the case of CLSE-Span and CoLoSS, the nonlinear LFPCs exhibit very low correlations. Mean correlations across all tasks are also very low; the direction is strongly influenced by CLSE-Time and CLSE-Dual.

Judging from the results in Table 5.6, 80% of the top-ranked features for CLSE-Dual refer to the prosody in speech, more specifically, they belong to the duration-based features. The remaining features indicate the importance of the Teager energy operator (nonlinear time-domain LFPCs). The two best absolute

Table 5.5 Top ten features ranked by absolute correlation for CLSE-Time

		CC_S				
Rank	CLSE-Time features	T	D	S	C	Mean
1	NFD-LFPC 11 (Q1)	0.748^{**}	0.342^{**}	-0.106^{**}	-0.041^{**}	0.236
2	NTD-LFPC 7 (Mean)	0.725^{**}	0.420^{**}	0.007	−0.029	0.281
3	NTD-LFPC 7 (Q2)	0.723^{**}	0.432^{**}	0.008	−0.003	0.290
4	NTD-LFPC 8 (Mean)	0.718^{**}	0.440^{**}	0.048	−0.018	0.297
5	NFD-LFPC 11 (Mean)	0.716^{**}	0.333^{**}	-0.070^{**}	-0.035^{*}	0.236
6	NFD-LFPC 6 (Mean)	0.709^{**}	0.402^{**}	−0.042	−0.009	0.265
7	NTD-LFPC 2 (Mean)	0.708^{**}	0.422^{**}	0.011	−0.028	0.278
8	NTD-LFPC 2 (Q2)	0.707^{**}	0.423^{**}	0.015	−0.015	0.283
9	NFD-LFPC 8 (Mean)	0.705^{**}	0.391^{**}	0.008	−0.030	0.269
10	NTD-LFPC 4 (Mean)	0.705^{**}	0.409^{**}	0.002	−0.022	0.274

Significance denoted by $^{*}p < 0.05$, $^{**}p < 0.01$. Abbreviations: C (CoLoSS), CC_S (Spearman's correlation coefficient), D (CLSE-Dual), Q1 (25th percentile), Q2 (50th percentile), S (CLSE-Span), T (CLSE-Time)

Table 5.6 Top ten features ranked by absolute correlation for CLSE-Dual

		CC_S				
Rank	CLSE-Dual features	T	D	S	C	Mean
1	Silent pause frequency ratio	0.464^{**}	-0.746^{**}	0.053	0.047^{**}	−0.045
2	Silent pause frequency	0.471^{**}	-0.706^{**}	0.052	0.047^{**}	−0.034
3	Ph spk-rate (SD)	0.475^{**}	0.589^{**}	0.027	0.140^{**}	0.308
4	Syl duration (Max)	0.083	0.533^{**}	−0.012	0.078^{**}	0.170
5	Syl duration (Max − Mean)	0.095	0.495^{**}	−0.029	0.081^{**}	0.160
6	NTD-LFPC 4 (Q3)	0.105	0.493^{**}	−0.011	-0.082^{**}	0.126
7	Mean silent pause duration	0.339^{**}	0.482^{**}	0.049	0.055^{**}	0.231
8	Speech segment length (Max)	0.079	0.478^{**}	0.032	−0.003	0.146
9	Syl duration (SD)	0.094	0.477^{**}	−0.014	0.061^{**}	0.154
10	NTD-LFPC 2 (Q3)	0.170^{*}	0.474^{**}	−0.008	-0.072^{**}	0.141

Significance denoted by $^{*}p < 0.05$, $^{**}p < 0.01$. Abbreviations: C (CoLoSS), CC_S (Spearman's correlation coefficient), D (CLSE-Dual), Ph (phoneme-based), Q3 (75th percentile), S (CLSE-Span), SD (standard deviation), spk-rate (speaking rate), Syl (syllable-based), T (CLSE-Time)

correlations are obtained by silent pause features. Interestingly, both features exhibit a strong negative correlation showing the opposite effect compared to the other tasks and the scientific literature (cf. Section 2.3.2.2). The reason for this may be attributed to delays in subjects' responses due to more intense cognitive load under time pressure, which, in turn, can result in fewer pauses between the subsequent responses. As expected, a positive low up to a moderate correlation can be observed for the duration of syllables (maximum and maximum-mean difference) and for the mean silent pause duration. In addition, the variation in speech tempo is found to be relevant, which is reflected by the standard deviation of the phoneme speaking rate and syllable duration. The former also contributes to the highest mean correlation across all tasks.

In Table 5.7, the most relevant features are shown for CLSE-Span. It can be seen that the overall best feature is the onset latency. Surprisingly, the (significant) correlation value indicates the opposite effect compared to the scientific literature (Section 2.3.2.2). Although it is not clear if the audio segments of CLSE-Span correspond exactly to the log-data from recording sessions, one can assume that there is a common position regarding the stimulus, most probably at the starting point of an audio segment. For the remaining tasks, the onset latency provides no valid results, because it refers to the first stimulus as well as the first response (cf. Section 4.2.3.2), whereas each trial in CLSE-Time and CLSE-Dual consists of many stimuli and responses; in the case of CoLoSS, stimulus information was deliberately excluded by trimming (cf. Section 3.2.3). Looking at the ranks

Table 5.7 Top ten features ranked by absolute correlation for CLSE-Span

		CC_S				
Rank	CLSE-Span features	T	D	S	C	Mean
1	Onset latency	-0.210^{**}	0.392^{**}	-0.451^{**}	0.034^{*}	−0.059
2	F_0 (Regr. intercept)	-0.308^{**}	0.077	-0.213^{**}	−0.023	−0.117
3	Intensity (Pos of max)	0.201^{*}	0.189^{*}	-0.206^{**}	0.014	0.050
4	NTD-LFPC 3 (Pos of max)	0.147	0.310^{**}	-0.195^{**}	0.034^{*}	0.074
5	NFD-LFPC 12 (Q1)	0.651^{**}	0.292^{**}	-0.191^{**}	-0.037^{*}	0.179
6	NTD-LFPC 9 (Pos of max)	0.013	0.352^{**}	-0.187^{**}	0.019	0.049
7	MFCC 1 (SD)	0.090	−0.022	0.181^{**}	−0.030	0.055
8	MFCC 1 (Q3)	-0.444^{**}	−0.151	0.181^{**}	0.023	−0.098
9	NTD-LFPC 5 (Pos of max)	0.184^{*}	0.266^{**}	-0.172^{**}	0.003	0.070
10	NFD-LFPC 3 (Pos of max)	0.322^{**}	0.105	-0.171^{**}	−0.012	0.061

Significance denoted by $^{*}p < 0.05$, $^{**}p < 0.01$. Abbreviations: C (CoLoSS), CC_S (Spearman's correlation coefficient), D (CLSE-Dual), Pos (position), Q1 (25th percentile), Q3 (75th percentile), S (CLSE-Span), SD (standard deviation), T (CLSE-Time)

two and three, prosodic features are found to be relevant—in particular, the simple linear regression intercept of F_0 and the relative position of the maximum intensity. Further, five TEO-based features (nonlinear LFPCs) and two spectral features (MFCCs) are among the top ten. It is noticeable that 50% of the most relevant features are related to the relative position of the global maximum. However, apart from the onset latency, the correlation values for CLSE-Span are very low.

The ten most relevant features for CoLoSS are summarised in Table 5.8. The overall best CoLoSS feature is the utterance duration which, however, shows insignificant results for CLSE-Span. One can see that almost all features refer to the prosody. With 80%, tempo in speech (speaking rate, articulation rate, and durations) based on phonemes forms the majority proportion among the top ten features. Some of them are particularly relevant for CLSE-Time: a moderate correlation can be observed for the minimum as well as the maximum-mean difference of the speaking rate. An opposite effect regarding the average speaking rate is obtained on CLSE-Dual; the reasons for this are stated above. Moreover, the minimum of the phoneme-based speaking rate is found to be relevant but does not generalise to CLSE-Dual and CLSE-Span. One of the top CoLoSS features refers to the shape of the intensity time series (simple linear regression slope), which, in addition, yields an almost moderate correlation for CLSE-Time. Apart from that, the TEO feature NTD-LFPC 12 (simple linear regression slope)—ranked as number ten—shows a very low (positive) correlation; its results are statistically significant across all tasks. Finally, although the highest absolute,

Table 5.8 Top ten features ranked by absolute correlation for CoLoSS

		CC_S				
Rank	CoLoSS features	T	D	S	C	Mean
1	Utterance duration	0.434^{**}	0.345^{**}	0.034	0.284^{**}	0.274
2	Ph spk-rate	-0.442^{**}	0.471^{**}	−0.021	-0.229^{**}	−0.055
3	Ph spk-rate (Min)	-0.606^{**}	0.000	0.002	-0.229^{**}	−0.278
4	Intensity (Regr. slope)	0.497^{**}	0.135	−0.053	0.179^{**}	0.190
5	Ph art-rate	−0.136	−0.118	−0.020	-0.167^{**}	−0.110
6	Ph duration (SD)	0.359^{**}	0.062	0.002	0.156^{**}	0.145
7	Ph duration (Mean)	0.198^{*}	0.028	0.016	0.152^{**}	0.098
8	Ph duration (Mean − Min)	0.146	0.013	0.007	0.151^{**}	0.079
9	Ph spk-rate (Max − Mean)	0.518^{**}	0.308^{**}	0.032	0.149^{**}	0.252
10	NTD-LFPC 12 (Regr. slope)	0.522^{**}	0.248^{**}	-0.100^{**}	0.146^{**}	0.204

Significance denoted by $^{*}p < 0.05$, $^{**}p < 0.01$. Abbreviations: art-rate (articulation rate), C (CoLoSS), CC_S (Spearman's correlation coefficient), D (CLSE-Dual), Ph (phoneme-based), S (CLSE-Span), SD (standard deviation), spk-rate (speaking rate), T (CLSE-Time)

mean correlation across all task is given by the minimum of the phoneme-based speaking rate, there is only little if any correlation in the case of CLSE-Dual and CLSE-Span.

5.2.2 Cross-Corpus Feature Ranking

For cross-corpus feature ranking, all four corpora were initially agglomerated on instance-level resulting in a new *large* dataset. The features were then ranked according to the absolute correlation using the new dataset.

Table 5.9 depicts the ten most relevant cross-corpus features by absolute correlation and shows their importance for each corpus individually. As can be seen in the table, only prosodic features are among the top ten, eight of which belong to the duration-based features. All ten cross-corpus correlations are statistically significant, but the strength is generally very low. The highest correlation is obtained by the utterance duration, though it shows weak results for CLSE-Span. Four phoneme-based speaking rate features are found to be particularly relevant (ranks two to five). In this connection, the correlation direction regarding the variation of speaking rates (SD and max-mean difference) is positive across tasks—the results are statistically significant except for CLSE-Span. For the minimum and mean of the phoneme-based speaking rate, cross-corpus results are only consistent with those of CLSE-Time and CoLoSS. Besides duration-based features, the simple linear regression slope of the intensity and F_0 indicates the importance of the global progression regarding prosodic low-level

Table 5.9 Top ten features ranked by absolute correlation across all corpora

		CC_S				
Rank	Cross-corpus features	T	D	S	C	Cross
1	Utterance duration	0.434^{**}	0.345^{**}	0.034	0.284^{**}	0.202^{**}
2	Ph spk-rate (Min)	-0.606^{**}	0.000	0.002	-0.229^{**}	-0.148^{**}
3	Ph spk-rate (Mean)	-0.442^{**}	0.471^{**}	−0.021	-0.229^{**}	-0.138^{**}
4	Ph spk-rate (SD)	0.475^{**}	0.589^{**}	0.027	0.140^{**}	0.129^{**}
5	Ph spk-rate (Max − Mean)	0.518^{**}	0.308^{**}	0.032	0.149^{**}	0.125^{**}
6	Onset latency	-0.210^{**}	0.392^{**}	-0.451^{**}	0.034^{*}	-0.123^{**}
7	Ph art-rate	−0.136	−0.118	−0.020	-0.167^{**}	-0.110^{**}
8	Syl spk-rate (Mean)	-0.422^{**}	−0.119	−0.026	-0.139^{**}	-0.109^{**}
9	Intensity (Regr. slope)	0.497^{**}	0.135	−0.053	0.179^{**}	0.109^{**}
10	F_0 (Regr. slope)	0.620^{**}	0.081	0.047	0.118^{**}	0.107^{**}

Significance denoted by $^{*}p < 0.05$, $^{**}p < 0.01$. Abbreviations: art-rate (articulation rate), C (CoLoSS), CC_S (Spearman's correlation coefficient), D (CLSE-Dual), Ph (phoneme-based), S (CLSE-Span), SD (standard deviation), spk-rate (speaking rate), Syl (syllable-based), T (CLSE-Time)

feature sequences, though the correlations are not significant for CLSE-Dual as well as CLSE-Span.

5.3 Information Gain

The relationship between single features and the level of cognitive load, measured by the correlation, is discussed in Section 5.2. Even when the correlation is very weak due to a non-monotonic relationship, one can still obtain a high degree of information about the label produced by the feature. In the ongoing, the order of cognitive load levels is not decisive for feature relevance analysis; the information shared by the feature and the label is of interest. The basis for this type of measurement is provided by the concept of entropy.

Entropy is an information-theoretic measure of *uncertainty* of a random variable (Cover and Thomas, 1991, p. 12). The entropy $H(Y)$ of a discrete variable Y is measured in bits of information and is given by (Cover and Thomas, 1991, p. 13):

$$H(Y) = -\sum_{y \in Y} p(y) \log_2 p(y). \tag{5.2}$$

The value $p(y)$ represents the probability of the individual class y. It is calculated as the number of occurrences of y divided by the total number of instances in the

dataset. By introducing a feature X, the values of Y in the dataset are partitioned into subsets according to the individual values of X. The Entropy of Y after observing X is written as (Cover and Thomas, 1991, p. 16):

$$H(Y|X) = -\sum_{x \in X} p(x) \sum_{y \in Y} p(y|x) \log_2 p(y|x). \tag{5.3}$$

If the entropy of Y after observing X is less than the prior entropy $H(Y)$, then the feature can reduce uncertainty by providing information about Y. If a feature cannot reduce uncertainty, it must be irrelevant (Liu and Motoda, 1998, p. 26). In this context, the information gain (IG) of a feature X is defined as the difference between the prior entropy $H(Y)$ and the expected posterior entropy by X (Liu and Motoda, 1998, p. 26):

$$IG(X) = H(Y) - H(Y|X). \tag{5.4}$$

As a consequence, the feature with the highest IG has the lowest conditional entropy.

Both discrete features and discrete labels are required for IG computation. Regarding the CLSE database (Section 3.1), categorical labels already exist. In the case of CoLoSS, the discretised version of the cognitive load labels is considered (see Section 3.2.5). As in the feature relevance analysis by correlation (Section 5.2), all features of the CL-Extended feature set (Section 5.1.1) are taken into account for which the inter-speaker variability and environmental mismatch are minimised by applying speaker normalisation for each cognitive load task individually using z-score transformation ($\mu = 0, \sigma = 1$). Since subject information is not included in the test partition of the CLSE database (cf. Section 3.1.2), only the training and development set are used.

In this thesis, the IG is computed using the data mining toolkit WEKA 3 (version 3.8.1) (Witten et al., 2016). For feature discretisation, WEKA applies the minimum description length method (Fayyad and Irani, 1993) by default. Since the probability of individual classes varies across the datasets used in this thesis, different entropy values are obtained: 1.585 (CLSE-Time), 1.585 (CLSE-Dual), 1.550 (CLSE-Span), and 0.937 (CoLoSS). In the following sections, a broad perspective of feature relevance is given—different feature subsets are evaluated using the mean IG.

5.3.1 Inter-Feature Group Comparison

A central question in the automatic assessment of cognitive load from speech is what aspects of speech are most suitable for which task. To this end, the relevance of feature groups was investigated by the mean information gain (IG) for each corpus.

Table 5.10 Feature group relevance by mean information gain (IG) for each corpus

	Mean IG			
Feature Group	CLSE-Time	CLSE-Dual	CLSE-Span	CoLoSS
Prosodic	0.100	0.042	0.004	0.013
Spectral	0.015	0.006	0.002	0.000
Voice quality	0.027	0.010	0.001	0.000
TEO	0.054	0.024	0.002	0.001

The results are summarised in Table 5.10. One can see that the prosodic features are superior across all four corpora. The second best feature group includes TEO-based features, except for CLSE-Span. All groups are by far most effective for CLSE-Time. In the case of CoLoSS, there is no relevance regarding spectral and voice quality features.

Due to the inclusion of all features per feature group, there is an overwhelming majority of features with a low IG. This, in turn, has a negative effect on the mean IG, whereby individual aspects within a group are not taken into account. In order to compensate for this fact, intra-feature group measurements were carried out (see Section 5.3.2).

Alternatively, feature groups can be analysed using a relevant subset of features. This approach was followed—the top 100 features that show highest IG were selected for each corpus individually (for details, see Appendix A.3). Furthermore, mixtures of tasks were considered by performing feature ranking on fused corpora (pooled on instance-level). In the ongoing, any combination of tasks is denoted as 'cross-corpus'. Figure 5.1 summarises the obtained feature subsets by the share of each feature group. Another aspect to which considerable importance should be attached is also illustrated, namely the shares of functional groups.

Now, the feature subsets are compared to the full feature set (CL-Extended). In this connection, a group of features is considered to be particularly relevant for a corpus (or cross-corpus) if its share among selected features is larger than its share of the full feature set. Notable differences in the importance can be observed for the feature groups. Interestingly, each corpus and cross-corpus is dominated by TEO-based features, especially in those cases where tasks are performed under stressful conditions (i.e., CLSE-Time and CLSE-Dual). For the CoLoSS corpus as well as for each cross-corpus in conjunction with CoLoSS, prosodic features are particularly relevant, but the major proportion is still given by TEO features. The feature subset obtained by CLSE-Span, also by its combination with CLSE-Time and CLSE-Dual, shows a similar proportion of feature groups in comparison with that of the full feature set. Concerning the functional groups,

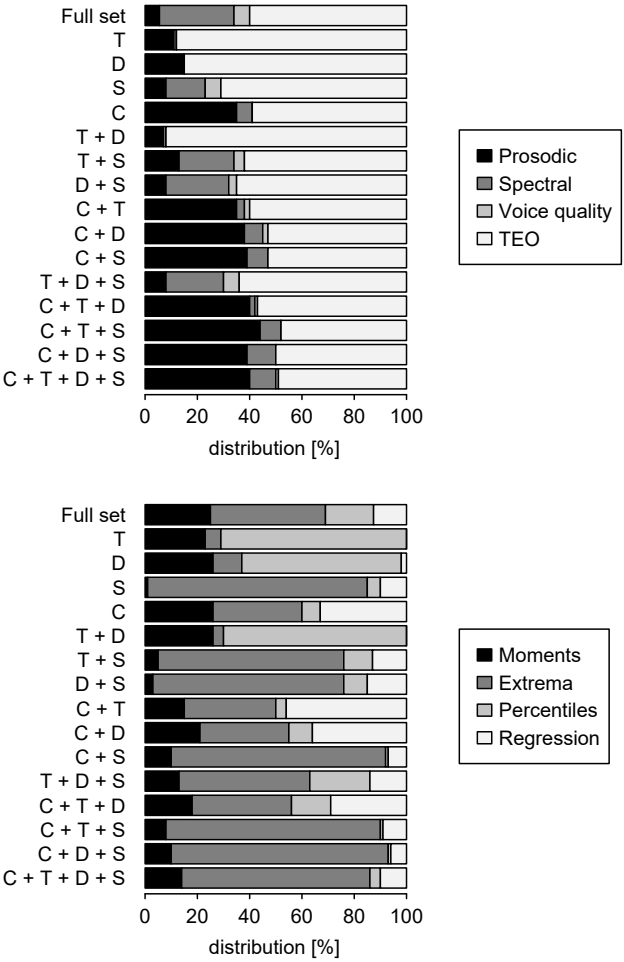


Figure 5.1 Feature relevance by feature group (above) and functional group (below). Distributions in % of the top 100 features, ranked by the information gain for each cognitive load corpus and cross-corpus, are shown and compared to the full feature set (CL-Extended). Corpora: T (CLSE-Time), D (CLSE-Dual), S (CLSE-Span), C (CoLoSS)

percentiles seem to be promising for both CLSE-Time and CLSE-Dual. On the other hand, extrema are particularly relevant for CLSE-Span, also in cross-corpus settings. It can also be seen that regression functionals are important for the CoLoSS corpus and for its combination with CLSE-Time as well as CLSE-Dual.

5.3.2 Intra-Feature Group Comparison

Feature group comparison, presented in the previous section, provided a rather broader view in terms of feature relevance. Now, feature types associated with the groups will be examined closer. To this end, the mean information gain (IG) was computed for the 13 different feature types of CL-Extended (cf. Section 5.1.1).

Table 5.11 shows the results for each corpus individually. Looking at CLSE-Time and CLSE-Dual, the intensity is the most relevant prosodic feature type and, moreover, the most important feature type across all groups. For the CoLoSS corpus, the duration-based features are superior to all others. Regarding CLSE-Span, mean IG values are identical in the prosodic as well as spectral feature group. The best spectral feature type for the remaining three corpora is the spectral centroid followed by formants and MFCCs. Next, the table reveals that the voice quality features jitter and cepstral peak prominence (CPP) are particularly relevant for CLSE-Time. The latter also seems to be promising for CLSE-Dual. With regard to CoLoSS, however, voice quality and spectral

Table 5.11 Feature type relevance by mean information gain (IG) for each corpus

	Mean IG			
Feature Type	CLSE-Time	CLSE-Dual	CLSE-Span	CoLoSS
Prosodic features				
Intensity+Δ+ΔΔ	0.156	0.068	0.004	0.001
F_0+Δ+ΔΔ	0.038	0.000	0.004	0.001
Duration	0.104	0.052	0.004	0.026
Spectral features				
Spectral centroid+Δ+ΔΔ	0.053	0.029	0.002	0.001
12 MFCC+Δ+ΔΔ	0.012	0.004	0.002	0.000
Formants $F_{1-3}(f,b)$+Δ+ΔΔ	0.016	0.006	0.002	0.000
Voice quality features				
Jitter+Δ+ΔΔ	0.036	0.004	0.000	0.001
Shimmer+Δ+ΔΔ	0.019	0.009	0.000	0.000
HNR+Δ+ΔΔ	0.015	0.002	0.001	0.000
CPP+Δ+ΔΔ	0.037	0.024	0.002	0.000
TEO-based features				
16 TEO-CB-Auto-Env+Δ+ΔΔ	0.020	0.004	0.001	0.000
12 NTD-LFPC+Δ+ΔΔ	0.085	0.044	0.003	0.002
12 NFD-LFPC+Δ+ΔΔ	0.069	0.029	0.002	0.002

features show hardly any relevance. Finally, the feature types of the TEO group are analysed. It can be observed that nonlinear LFPCs (NTD-LFPCs and NFD-LFPCs) are the most relevant TEO features across all four corpora. In three out of four cases, NTD-LFPCs yield highest mean IG values.

Chapter 6

Recognition Experiments

The main focus of this thesis is on the assessment of cognitive load from speech. Feature relevance analyses, presented in Chapter 5, revealed the effectiveness of single features and feature sets for different cognitive load tasks. In this chapter, speech features are evaluated and discussed with regard to automatic cognitive load recognition. First, in Section 6.1, fundamental prerequisites for carrying out the experiments are presented. The experimental methodology used throughout the experiments is described in Section 6.2. A series of experiments is conducted with the aim of verifying the suitability of relevant speech features for various cognitive load scenarios. It starts with a systematic evaluation of features for cognitive load classification with tasks considered in isolation (Section 6.3). In Section 6.4, feature sets are compared in terms of the generalisation capability of models, which is determined by cross-corpus classification of cognitive load. This is followed by the evaluation of feature sets for classification scenarios in which data from different tasks including the target task are taken into account for modelling (Section 6.5). Next, classification experiments are conducted according to the rules of the INTERSPEECH 2014 COMPARE challenge in Section 6.6, whereby the results obtained by features and algorithms used in this thesis are compared with those of the official baseline system. Finally, regression-based approaches to cognitive load recognition are introduced and evaluated in Section 6.7.

6.1 Requirements, Measures, and State of the Art

In the following sections, various methods and measures are introduced that constitute the basis for the experiments in this thesis. In addition, a summary of existing automatic speech-based cognitive load classification systems is given.

6.1.1 Resampling

Real-world data is usually unbalanced in terms of the class distribution, which can have negative effects on the learning performance of a classifier (Sotiropoulos and Tsihrintzis, 2017). Resampling techniques focus on changing the number of instances in the training data to adjust the distribution among classes. Generally, it is not clear beforehand which technique is most suitable regarding the classifier's performance. Some of the most popular solutions are described in the following.

The most straightforward solution is to collect more data from the minority classes in order to be included in the training data, also known as *natural resampling*. However, this approach is associated with high effort, because, as stated above, real-world data is principally unbalanced. The simplest method is certainly to increase the minority classes through random replication of the corresponding instances. This technique is referred to as *random over-sampling*. The counterpart, also known as *random under-sampling*, aims at balancing the training data through the random removal of instances belonging to the majority class. There are also techniques that perform instance generation instead of replication to address the class imbalance problem. The most popular of such techniques is called SMOTE (Synthetic Minority Over-sampling Technique) (Chawla et al., 2002). The main idea behind SMOTE is to create synthetic examples by interpolating between several neighbouring instances of the minority class.

Apart from changing the class distribution, some other strategies have been proposed as a remedy for the class imbalance problem on feature-level and classifier-level. For further information, a summary of solutions is given by Guo et al. (2008).

6.1.2 Feature Normalisation

Feature normalisation is a common requirement for most learning algorithms. It aims at adjusting feature values measured on different scales to a common scale and provides a way to compensate the environmental mismatch between different datasets. Normalisation can result in faster training time and better model accuracy (Priddy and Keller, 2005, p. 15). In audio processing, feature normalisation can be carried out on different levels as described in the following sections.

6.1.2.1 Normalisation of Low-Level Features

The normalisation of low-level features is commonly applied in the field of automatic speech recognition (e.g., Liu et al., 1994; Viikki and Laurila, 1998; Zolnay et al., 2005). In this connection, cepstral features, represented by a sequence of feature vectors, are normalised per utterance with regard to the mean

and the variance. *Cepstral mean normalisation* aims at removing the average from the feature values, whereas *cepstral variance normalisation* scales feature values to have a unit variance (Gales and Young, 2008). Both methods can be combined to *cepstral mean and variance normalisation*, which has been shown to be a robust approach against additive noise (Strand and Egeberg, 2004).

Speaker normalisation is another approach, which can be used in low-level processing stages. Speakers have vocal tracts of different sizes, which, in turn, causes formants to shift in frequency. In order to compensate for this fact, the frequency axis in filterbank analysis can be adjusted accordingly. This method is also known as *vocal tract length normalisation* (Young et al., 2006).

Among the approaches for normalising low-level features, a method called *feature warping* has gained interest, which maps the feature distribution to the standard normal distribution over a specific time interval (Pelecanos and Sridharan, 2001). This method was also applied on a per speaker basis—initially proposed for emotion recognition (Sethu et al., 2007).

6.1.2.2 Normalisation of Statistical Features

In the context of static modelling (Section 6.1.4), statistical functionals can be applied to low-level features (cf. Section 4.6.2) and the resulting feature vector can then be normalised before feeding it into a learning machine—which is the preferred approach in this thesis. It should be noted that feature processing using the concept of functionals is, in addition, a sort of normalisation over time, because a single feature vector with a constant number of elements is obtained regardless of the segment duration (Schuller et al., 2011a). In fact, segments vary in length due to the nature of speech: words differ in duration because of different numbers of phonemes and the way words are expressed by the speaker; in turn, the length of an utterance is influenced by combinations of these substructures.

Typical transformations of derived statistics include the *min-max normalisation* and the *z-score normalisation*. In the case of min-max normalisation, the data is scaled to a fixed range—usually between 0 and 1. This technique computes normalised values of a feature x by (Berthold et al., 2010, p. 130):

$$x' = \frac{x - min_x}{max_x - min_x}, \tag{6.1}$$

where min_x and max_x represent the minimum and maximum of all observed values, respectively. The z-score normalisation, also known as standardisation, rescales features in the way that they have a mean value of 0 and a standard deviation of 1. The z-scores of a feature x are computed by (Berthold et al., 2010, p. 130):

$$x' = \frac{x - \mu_x}{\sigma_x}, \tag{6.2}$$

where μ_x and σ_x are the mean and standard deviation of all observed values, respectively.

The min-max normalisation is very sensitive to outliers in the sense that a single outlier can force the whole data to concentrate only in a small interval. As this effect is less prominent for z-scores, z-score normalisation is preferred for the experiments in this thesis.

In addition, there is the question of what data is used to compute the normalisation parameters and on what data the normalisation is performed. In this regard, the chosen normalisation method influences, among other factors, the prediction performance of a model (cf. Schuller et al., 2010, 2014). In the following, four methods are defined that are investigated in the experiments of this thesis. The first of these methods is denoted as *partition normalisation* (PN). The PN method normalises features of training and test partitions individually. This method is typically used if sufficient amount of representative data is available for each partition. However, test sets are sometimes too small to compute reliable scaling parameters or systems are conceptualised for online data processing, where each test instance is processed ad-hoc without considering information from the whole test set. In such cases, normalisation parameters, applied to both training and test data, are computed only from the training data. In the ongoing, this method is referred to as *training normalisation* (TN). Next, *speaker normalisation* (SN) is used to remove the inter-speaker variability. This is realised by scaling the feature values on a per speaker basis. Usually, the recording conditions differ across speech corpora; for example, there are varying room acoustics, microphone types, and microphone distances. In order to eliminate this environmental mismatch, features can be normalised for each corpus individually, hereinafter referred to as *corpus normalisation* (CN). This method is assumed to be particularly useful for cross-corpus experiments.

6.1.3 Feature Selection

The success of machine learning is based on many factors. Particularly crucial for this field are the properties of feature sets designed for modelling. The occurrence of irrelevant or redundant features may have negative effects on learning algorithms (Liu and Motoda, 1998, p. 19). Feature selection is the process of selecting a subset of relevant features for model construction. The utilisation of feature selection techniques provides a variety of advantages:

- Less data allowing the learning algorithms to operate faster,
- Higher accuracy and better generalisation from data, if the right subset is chosen,
- Simpler and more compact representation, which is easier to understand, and
- Fewer features have to be extracted regarding new data from the same domain.

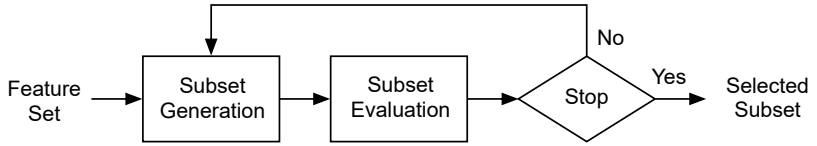


Figure 6.1 A unified view of feature selection

Basically, feature relevance depends on the type of the evaluation measure. For example, if accuracy is the measure of interest and a feature's removal causes a reduction in the classifier's accuracy, then this feature is relevant (Liu and Motoda, 1998, p. 29). In addition, Kohavi and John (1997) have made a distinction between strong and weak relevance. Strong relevance means that the feature is always necessary. If this feature is removed the results will deteriorate. Weak relevance stands for a feature, which is not always necessary, but it may become necessary under certain conditions. Besides the issue of feature relevance, high-dimensional data often contain many redundant features. It is widely accepted that perfectly correlated features are redundant. Redundant features can be seen as a type of irrelevant features; the removal of one of them will not affect the learning performance (Liu and Motoda, 2007, p. 5).

Feature selection is not the same procedure as feature reduction. Feature reduction (or feature extraction) techniques, such as principal component analysis, transform features from the original feature space into another space resulting in fewer features, whereas feature selection techniques operate within the same feature space. One can differentiate between supervised and unsupervised feature selection. In supervised feature selection—the preferred selection method in the experiments of this thesis—, labels are used to measure the goodness of features. In the case of unsupervised feature selection, where labels are not present, a common approach is the method of clustering. For further information on unsupervised feature selection, the reader is referred to Liu and Motoda (2007).

The aspects of feature selection can be generalised into a unified framework as depicted in Figure 6.1. The following three stages constitute the search for an optimal feature subset: subset generation, subset evaluation, and stopping criteria. The search is repeated as long as predefined stopping criteria are not fulfilled. Typical criteria are, for example, the goodness of the evaluation result, the number of minimum features needed, or the completion of the overall search. Subset generation defines the search strategy including search direction and heuristic. At each time of the feature selection process, a new subset is generated and, subsequently, evaluated using an evaluation measure. In the following sections, two aspects of feature selection are discussed in more detail: the search problem and feature evaluation.

6.1.3.1 The Search Problem

Feature selection can be regarded as a search problem, where feature subsets are represented by a state in the search space (Liu and Motoda, 1998). The optimal subset of features can typically be found by using exhaustive search. To accomplish this with n features, 2^n combinations have to be evaluated, which is, however, not feasible for a large number of features. The search through the space of feature subsets is linked to two issues, namely the search direction and the search strategy.

The *forward generation* (or forward selection) begins with an empty set of features. In the search phase, features are added to the feature set based on some criteria. The amount of selected features grows until it reaches the amount of the original feature set. In contrast, the *backward generation* (or backward elimination) begins with the full set of features. Here, the least important feature is removed—the feature set shrinks until there is only one feature available. One can also take the advantage of both directions, also known as *bidirectional generation*.

In order to avoid the time-consuming exhaustive search, heuristics can be applied by using graph-based search algorithms. Heuristic search is much faster because it only searches a specific path and finds an approximately optimal subset—it does not guarantee that the found feature subset is optimal. A frequently encountered heuristic search is the best-first search strategy. Best-first search expands its search space layer by layer. All child nodes of a currently chosen node represent newly generated subsets. These child nodes are evaluated and the child node that produces the best result is chosen. Additionally, best-first search allows backtracking along the search path, which facilitates searching from a promising previous subset. The process of searching is repeated until no further expansion is available. It is common that a stopping criterion is used, such as the number of fully expanded nodes that do not contribute to an improvement. Alternatively, the greedy hill-climbing heuristic is a simpler strategy. It functions in a manner similar to the best-first search, but without backtracking. The search path will only be expanded if the result of the best performing child node is equal or greater than the currently chosen node. If there is no improvement, the algorithm stops and gives the approximately optimal subset located at this point.

6.1.3.2 Feature Evaluation

Regarding the way in which features are evaluated, there are two main approaches: wrappers and filters.

Wrappers use the performance of a learning algorithm for feature evaluation. For instance, the classifier's accuracy can be used directly as a measure. The idea behind this approach is that the feature selection algorithm exists as a wrapper

around the learning algorithm, i.e., the learning algorithm can be regarded as a black box. A popular search method for the wrapper approach is the so-called *sequential forward floating search* (Pudil et al., 1994). Once the optimal feature subset is found, the learned model can be evaluated on the independent test set. Usually, it makes no sense to use a different learning algorithm in the final testing phase, because the optimal feature subset is obtained by the interaction between a specific learning algorithm and the dataset.

The filter approach is based upon an independent measure for evaluating features without the utilisation of a learning algorithm. Typical measures refer to the information, distance, dependence, or consistency. In its simplest form, a filter evaluates each feature individually and performs feature ranking according to the evaluation results that reflect the degree of feature relevance. A subset of features can then be defined by the number of top-ranked features or a threshold concerning the relevance score.

In the experiments of this thesis, the following three filter-based feature selection methods, each based on the algorithms of the WEKA 3 data mining toolkit (Witten et al., 2016), are investigated:

- IG*N*: The information gain (IG) is used to measure the relevance of each feature individually. IG determines how much information about the labels is provided by the feature (cf. Equation (5.4)). The chosen criterion for feature selection is the number N of top-ranked features.
- $\mathrm{CC_P}N$: This method evaluates the worth of a feature by measuring the Pearson's correlation (cf. Equation (5.1)) between it and the classes after class binarisation (one-against-all) is performed. The top N features, ranked by the correlation coefficient (average of absolute, binary-class correlations), are selected.
- CFS: The correlation-based feature selection (CFS) algorithm (Hall, 1999) measures the goodness of feature subsets. The core idea behind this method is to find a feature subset in which each feature is highly correlated with the labels while the correlation of the features among each other is low. As a consequence, this method takes both feature relevance and feature redundancy into account. The CFS algorithm is applied in conjunction with the forward best-first search strategy.

Generally, the wrapper models lead to better results due to the inclusion of the learning algorithm during the selection process. However, feature selection using the filter approach is faster, because measures such as correlation or information gain are usually cheaper than measuring the accuracy of a classifier. In cases where learning algorithms cannot handle large-scale feature sets, filters can be applied as a pre-processing step to reduce the dimensionality. Moreover, the filter approach provides a kind of generic feature selection (independent of the learning

algorithm), meaning that filters can overcome overfitting (Guyon and Elisseeff, 2003).

6.1.4 Modelling

The aim of an audio analysis system is to interpret recorded data. For this purpose, features are extracted from labelled data and an appropriate model needs to be learned.[1] The feature vectors and the labels are used to build either classification or regression models. While classification models assign categorical values to the test instances, regression models produce continuous numerical values. Before applying an analysis system to real-world use-cases, it is common practice to first simulate the system. To this end, specific datasets are used for evaluation and optimisation.

Generally, one can distinguish between static and dynamic learning algorithms. Static learners operate on constant-dimensional feature vectors for time series of variable length. In this connection, acoustic low-level descriptors are first processed using statistical functionals (Section 4.6.2) or bag-of-audio-words (Bhatia et al., 2017). More recently, methods that operate directly on the raw time representation of a signal have gained attention (Trigeorgis et al., 2016). Moreover, it has been shown that convolutional neural network architectures, mainly developed for analyses of images, can be used to process spectrograms of an audio signal (Weißkirchen et al., 2017). However, static algorithms are not capable of modelling time-dependent aspects contained in the data. Dynamic learning algorithms, on the other hand, consider the time series directly. Such algorithms handle different tempo deviations—in fact, this is an important aspect especially in the field of automatic speech recognition, because spoken units such as vowels and consonants vary in time. The standard tool for modelling time-varying acoustic events is the Hidden Markov Model (HMM). In HMM-based acoustic models, each class is represented by an HMM that includes transition probability parameters a_{ij} and output observation distributions b_j. At each step in time, a transition is made based on the probabilities a_{ij} from the current state s_i to one of its connected states s_j (Gales and Young, 2008). The state sequence is not observable as such, i.e., it is hidden, but can be observed through the underlying sequence of observations (Rabiner, 1989).

Almost all works in the field of paralinguistic speech processing, including this thesis, perform static modelling on utterance-level (cf. Section 6.1.6; Anagnostopoulos et al., 2015). This can mainly be attributed to the nature of datasets. Speech corpora in this field usually contain a set of audio files, where each file rep-

[1] This approach belongs to the group of supervised machine learning methods since the model is build based on known outcome values. Unsupervised methods are not considered in this thesis.

resents an utterance that is described by a single label. The SEMAINE Database (McKeown et al., 2012), on the other hand, serves as an example where speech is continuously labelled: The raters could move a slider in a given range to rate their subjective opinion in terms of social interactions contained in audiovisual material; in this way, dimensional labels (e.g., *Valence*) were recorded at a rate of 50 samples per second.

The following sections give a short introduction to those static learning algorithms that are used for the experiments in this thesis.

6.1.4.1 Support Vector Machines

Support vector machines (SVMs) base on statistical learning theory and are primarily designed for binary, linear classification problems (Cortes and Vapnik, 1995). The fundamental idea behind SVMs is to find an optimal hyperplane between two classes that should generalise well. Figure 6.2 illustrates the construction of a hyperplane in a two-dimensional space for a typical linear separation problem. It can be seen that both labelled clusters are well separated. The distance between the separating hyperplane and one of the (canonical) hyperplanes through the closest points is called the margin. The closest points—the so-called support vectors—determine the maximum margin and influence the position of the separating hyperplane. This means that only a small amount of training data has to be taken into account for optimal separation. For the classification task, the data points x_i are differentiated between *positive* and *negative* according to their class assignment, i.e., data points are labelled as $y_i = +1$ for class 1, whereas those of class 2 are labelled as $y_i = -1$. The separating hyperplane is given as $w \cdot x + b = 0$, where w is the weight vector (normal vector to the hyperplane), x is the input vector, and b is the bias of the hyperplane. In the training phase, the parameters w and b are determined. The mathematical background behind SVMs is given by Cortes and Vapnik (1995).

For optimisation, the hyperparameter C of an SVM classifier can be adjusted. It defines its complexity and is also known as the softness parameter—larger values of C lead to smaller-margin hyperplanes.

So far, the classification for a linear separation problem has been described. In real-world conditions, however, the data of different classes could be partially mixed up, possibly with overlapping data points. The classification problem is then more complex and the data is not linearly separable. For such non-linear classification problems, the so-called *kernel trick* can be applied to map the data into a higher-dimensional space, where the classification problem can then be solved linearly. Frequently used kernel functions are the polynomial kernel, Gaussian kernel, and sigmoid kernel.

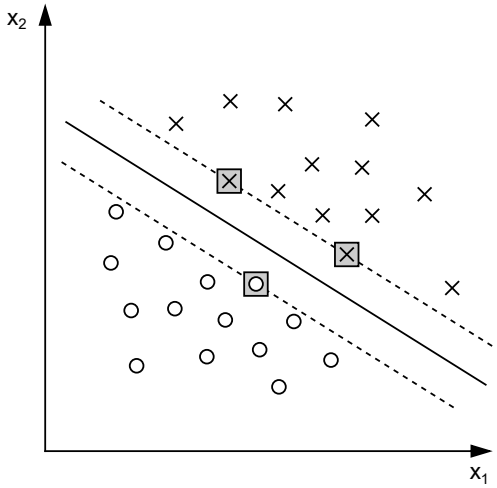


Figure 6.2 Example of an optimal hyperplane (solid line) in a two-dimensional space. The circles and crosses indicate instances belonging to a particular class. The support vectors are marked with grey squares and define the maximum margin of separation (parallel dashed lines) between the two classes

Although support vector machines are initially conceptualised for two-class problems, some strategies have been established to realise multi-class classification (Campbell and Ying, 2011, p. 8):

- A binary tree can be used, where the learning task is reduced to binary classification at each node. Depending on the outcome of a decision, one of the next two nodes is selected.
- A series of one-against-all classifiers can be used. In this case, separate SVMs are trained using data from a particular class as the positive labelled instances and the remaining classes are negatively labelled instances.
- One-class classifiers can be trained for each class. The idea is to construct a boundary around the *normal* data while data outside the boundary is classified as *abnormal*.

The success of SVMs is attributed to the fact that they are capable of handling large feature spaces, sparse features, and being robust against overfitting (Joachims, 1998). Moreover, the concept of support vector machines can be applied to regression problems, also known as support vector regression (SVR) (Drucker et al., 1997). The principle behind SVR is that constraints are used to allow for some errors within a certain distance to the true values.

6.1.4.2 Artificial Neural Networks

Inspired by the structure and function of the central nervous system of vertebrates, the organisation of biological neurons has been adopted in the field of computer science. The basis for artificial neural networks is provided by mathematical models that describe how neurons work (cf. McCulloch and Pitts, 1943; Rosenblatt, 1958). An artificial neural network can be regarded as a graph with a number of nodes and edges. The nodes in a network represent the neurons and the edges between them are the weighted synaptic connections (Jain et al., 1996).

Information processing of an artificial neuron can be described as follows: Each input variable x_i of a neuron j is multiplied by a weighted parameter w_{ji}. The weighted inputs are summed up together with a bias w_{j0}. This process results in an activation (Bishop, 2006, p. 227):

$$a_j = \sum_{i=1}^{N} w_{ji} x_i + w_{j0}. \tag{6.3}$$

The sum is then passed through an, usually non-linear, activation function, which gives the final output of the neuron.

There are different activation functions; the choice depends on the nature of the data and the assumed target variables. A frequently used activation function is the sigmoid function (Bishop, 2006, p. 228):

$$y(a_j) = \frac{1}{1 + e^{-a_j}}. \tag{6.4}$$

An alternative to the sigmoid function is the hyperbolic tangent, or tanh function (Bishop, 2006, p. 245):

$$y(a_j) = \frac{e^{a_j} - e^{-a_j}}{e^{a_j} + e^{-a_j}}. \tag{6.5}$$

For multi-class problems, the jth output, with $j = 1, \ldots, K$, of the last network layer can be normalised to determine the class probability. This is achieved by the softmax function (Bishop, 2006, p. 198):

$$y(a_j) = \frac{e^{a_j}}{\sum_{k=1}^{K} e^{a_k}}. \tag{6.6}$$

In the case of regression problems, where a single neuron is used in the last layer, the identity function $y(a_j) = a_j$ is typically the activation function of interest, since any continuous function can be approximated by networks (Bishop, 2006, p. 228).

Different types and topologies of artificial neural networks have been proposed over the years. A specific class of neural networks is the multilayer

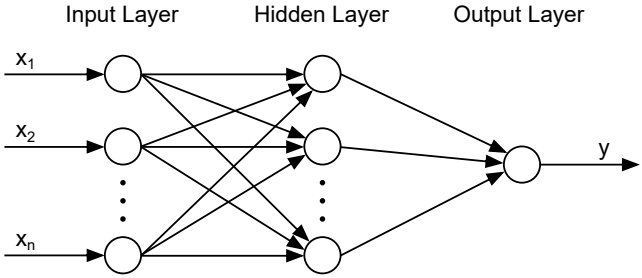


Figure 6.3 Example of a multilayer perceptron with one hidden layer and a single output neuron. The nodes represent the neurons and the arrows denote the information flow. The number of neurons in the input layer corresponds to the number of features. Connections between nodes indicate weight parameters

perceptron (MLP). MLPs belong to the group of feedforward neural networks. The neurons of an MLP network are arranged in at least three layers. The first layer takes the inputs, whereas the last layer produces the outputs. The layers between them are referred to as hidden layers—they have no connections to the external world. In principle, hidden layers extract features from the input space and can thus provide more appropriate representations. Each neuron of an input or hidden layer is connected to the neurons of the next layer. There is no backward connection or a connection among neurons in the same layer. Figure 6.3 gives an example of an MLP with one hidden layer and a single output neuron for binary classification or regression tasks.

The input weights of the neurons can be learnt to reduce the output error. Frequently used error functions are the mean square error or the cross-entropy. One of the most popular training techniques for multilayer networks is the gradient descent-based backpropagation algorithm (Rumelhart et al., 1986). It includes the following steps: (1) Propagate the input vector x_n forward through the network. (2) Compute the output error by taking the predicted and actual values into account. (3) Propagate the output error backwards from the final layer to the previous layers in order to obtain the errors for the hidden units.

Along with backpropagation, the gradient of error E is used to update the weights (Bishop, 2006, p. 240):

$$w^{(\tau+1)} = w^{(\tau)} - \eta \nabla E(w^{(\tau)}), \tag{6.7}$$

where τ labels the iteration step and the parameter $\eta > 0$ is known as the learning rate. For a deeper discussion and the mathematics behind network training, the reader is referred to Bishop (2006).

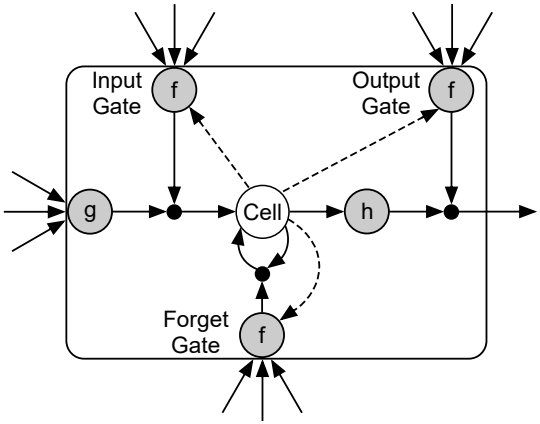


Figure 6.4 Long Short-Term Memory block with one cell. Inputs are the input data vector, previous outputs from cells in the current layer, and bias values. The gates ('f') represent non-linear summation units that collect the activations from inside and outside the memory block and control the activation of the cell via multiplicative units (black circles). Input and output activation functions are denoted as 'g' and 'h', respectively. Dashed lines indicate weighted peephole connections

Contextual information from the past can be added to a feedforward neural network by allowing cyclical backward connections. Such networks are called recurrent neural networks (RNNs). The recurrent connections of an RNN allow the inputs to persist in the hidden layers. However, for standard RNN architectures, the context can be quite limited in practice—the influence of a given input on the hidden layer, and thus on the network output, either decays or blows up exponentially over time. This effect is often referred to as the 'vanishing gradient problem' (Hochreiter et al., 2001). To overcome this limitation, the Long Short-Term Memory (LSTM) architecture (Hochreiter and Schmidhuber, 1997) was introduced. LSTM networks have shown remarkable performance in many tasks, for example, phoneme classification (Graves and Schmidhuber, 2005), handwriting recognition (Liwicki et al., 2007), and multimodal emotion recognition (Wöllmer et al., 2010). An LSTM layer consists of a set of recurrently connected memory blocks. Each block contains one or more self-connected memory cells and the following multiplicative units: the input, output, and forget gates. Figure 6.4 depicts an LSTM memory block with a single cell. The input and output of the cell are multiplied by the activations of the corresponding gates and the cell's previous state is multiplied by that of the forget gate. As a consequence, the network is capable of storing and accessing information over a long period

of time. For instance, the activation of the cell will not be overwritten by new inputs as long as the input gate is closed; it can be made available much later in the sequence by opening the output gate. The purpose of the forget gates is to reset memory cells. The 'peephole' connections improve the ability to learn tasks that require precise timing and counting of the internal states (Gers et al., 2002).

RNNs can be trained by the gradient descent-based backpropagation algorithm. Here, it is modified to BPTT—backpropagation through time (Werbos, 1990).

6.1.5 Evaluation Measures

The evaluation of the system's recognition performance requires an appropriate measure. The choice usually depends on some criteria. In the case of classification, the most common evaluation measure is the accuracy, also referred to as the recognition rate. It is defined as the ratio between the number of correctly classified test instances and the total number of test instances (Schuller, 2013, p. 133):

$$\mathrm{ACC} = \frac{\#\text{ correctly classified test instances}}{\#\text{ test instances}}. \tag{6.8}$$

The accuracy per class i, i.e., the class-specific recall R_i, can be used to compute the weighted average recall (Schuller, 2013, p. 133):

$$\mathrm{WAR} = \sum_{i=1}^{C} p_i R_i, \tag{6.9}$$

where p_i is the prior probability of class i and C is the number of classes. If the distribution of test instances among classes is unbalanced to a certain degree, the overall accuracy does not give a truthful performance measure. Instead, the unweighted average recall (UAR) can be selected (Schuller, 2013, p. 134):

$$\mathrm{UAR} = \frac{1}{C} \sum_{i=1}^{C} R_i. \tag{6.10}$$

Models that predict continuous numerical values are referred to as regression models. For evaluation: Let $x(n)$ and $y(n)$ be the predicted and actual values, respectively, with $n = 0, \ldots, N-1$, where N is the total number of instances. An accurate measure of the linear relationship between the two variables is provided by the Pearson's correlation coefficient $\mathrm{CC_P}$ (Equation (5.1)). However, if the data is non-normally distributed or we speak of ordinal variables, then the Spearman's correlation coefficient should be preferred (Field, 2013). The

Spearman's correlation coefficient CC_S is computed by first ranking the data, i.e., replacing the actual values of $x(n)$ and $y(n)$ by their ranks, and then applying the Pearson's equation.

In addition to the relationship between numerical variables, the model accuracy can be determined by considering the difference between the predicted values and the actual values. Common error measures are the mean absolute error (MAE), mean square error (MSE), and root mean square error (RMSE) (Kassambara, 2017, p. 60):

$$\mathrm{MAE} = \frac{1}{N}\sum_{n=0}^{N-1} |x(n) - y(n)|, \tag{6.11}$$

$$\mathrm{MSE} = \frac{1}{N}\sum_{n=0}^{N-1} \big(x(n) - y(n)\big)^2, \tag{6.12}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1} \big(x(n) - y(n)\big)^2}. \tag{6.13}$$

The MAE provides information about the average size of the prediction error and is not sensitive to error outliers. The MSE gives more weight to large deviations by taking the square of the errors. MSE provides useful information about the distribution and outliers, but it may warp the results—the interpretation should be done very carefully. To overcome this limitation, the square root of the MSE can be taken as given by the RMSE.

6.1.6 Existing Speech-Based Cognitive Load Classification Systems

Previous studies analysed the effects of cognitive load on speech parameters (Section 2.3.2) and have thus formed the basis for the development of speech-based cognitive load classification systems. Table 6.1 gives a summary of such systems—the authors, features, classification methods, and the recognition performance on the datasets are contained. The systems shown cover the following four tasks: Stroop test, reading comprehension, arithmetic task, and driving under cognitive load. All authors conducted speaker-independent experiments. Most of them used a leave-one-subject-out cross-validation strategy, i.e., data from each of the subjects is used exactly once as the test data while data of the remaining subjects is used for model training. Only the authors Boril et al. (2010) separated the total dataset into a single training partition and test partition. Probably the

Table 6.1 Summary of existing speech-based cognitive load classification systems

Reference	Features	Classifier	Task (C)	ACC [%]
Yin et al. (2007)	MFCC+ΔΔ, F_0+ΔΔ, intensity+ΔΔ	GMM	R (3)	71.10
Yin et al. (2008)	MFCC+ΔΔ, F_0+ΔΔ, intensity+ΔΔ	GMM	S (3)	77.50
Le et al. (2009)	Non-uniform filterbank, cepstral features	GMM	S (3)	76.50
Yap et al. (2009)	MFCC, F_0, intensity, phase based features	GMM	S (3)	85.30
Boril et al. (2010)	Cepstral features, spectral centroid	SVM Fusion of GMMs	D (2)	94.30
Gorovoy et al. (2010)	Tempo and pauses	Decision tree	A (3)	77.80
		MLP	A (3)	77.80
		Logistic regr.	A (3)	82.20
		Bayes net	A (3)	82.20
Le et al. (2010a)	Subband cepstral features	GMM	S (3)	80.00
Le et al. (2010b)	SMFCC, MFCC	GMM	S (3)	54.50
Yap et al. (2010a)	Regr. of formants $F_{1–3}$	GMM	S (3)	74.20
Yap et al. (2010b)	Formants $F_{1–2}$+Δ	GMM	S (3)	67.90
Le et al. (2011)	Spectral centroid features	GMM	S (3)	88.50
		GMM	R (3)	72.60
Yap et al. (2011a)	Formants $F_{1–3}$+Δ	GMM	S (3)	67.70
		GMM	R (3)	51.90
Yap et al. (2011b)	Formants $F_{1–3}$, CPP	GMM	S (3)	62.70
		GMM	R (3)	57.80

Task: A (arithmetic), D (driving), R (reading), S (Stroop test). Abbreviations: ACC (accuracy), C (number of classes)

first speech-based system for cognitive load recognition was introduced by Yin et al. (2007).

The table shows which speech features have been applied in the past. Some of the most effective features in almost any speech processing task are the well-known Mel Frequency Cepstral Coefficients (MFCCs). In the context of cognitive load classification, MFCCs have been extracted from the speech signal (cf. Yin

et al., 2007, 2008; Yap et al., 2009; Le et al., 2010b) and, additionally, from the glottal waveform output resulting in Source Mel Frequency Cepstral Coefficients (SMFCCs) (cf. Le et al., 2010b). Other works investigated cepstral features in conjunction with different filterbank configurations (cf. Le et al., 2009, 2010a). Apart from that, Boril et al. (2010) and Le et al. (2011) explored the spectral centroid, which represents the centre of gravity of the spectrum. The formants F_1, F_2, and F_3—commonly known as the first three resonance frequencies of the human vocal tract—were also used for cognitive load classification (cf. Yap et al., 2010b, 2011a,b). Yap et al. (2011b) suggested that the cepstral peak prominence (CPP) contains useful voice source information that complements the information captured by formants. Prosodic features have shown potential for the automatic classification of user states, for example, in speech emotion recognition (e.g., Dellaert et al., 1996; Luengo et al., 2005; Schuller et al., 2007). Such features also contribute to the classification of cognitive load. For instance, Gorovoy et al. (2010) used the articulation rate, pause rate, and pause percentage. Moreover, prosodic features such as the fundamental frequency F_0 and intensity were extracted from the speech signal (cf. Yin et al., 2007, 2008; Yap et al., 2009). In addition to the extraction of speech parameters, some works used information about their dynamics, for example, the first order derivative (Δ) (cf. Yap et al., 2010b, 2011a) and second order derivative ($\Delta\Delta$) (cf. Yin et al., 2007, 2008).

Table 6.1 indicates that most of the works employed a classification method based on Gaussian mixture models (GMMs), i.e., modelling a static distribution of features in the feature space with a series of Gaussian distributions. Furthermore, Boril et al. (2010) proposed a system, which uses a support vector machine (SVM) classifier in order to fuse the scores of various GMMs. Regarding the arithmetic task, Gorovoy et al. (2010) investigated the following four different classification methods: logistic regression, Bayes net, decision tree, and multilayer perceptron.

In summary, 88.50% accuracy is achieved for the Stroop test and 72.60% for the reading comprehension task. Both results were obtained by using spectral centroid features in connection with GMMs. For the arithmetic task, the highest accuracy with 82.20% could be achieved by tempo and pause features; for classification, both logistic regression and Bayes net were superior. Driving under cognitive load has been investigated as a binary classification problem with a combination of GMMs and SVM resulting in 94.30%. For further information on the systems, the reader is referred to the literature in Table 6.1. However, some of the systems are not easily comparable, since different speech databases were used.

As part of the conference INTERSPEECH 2014, the Computational Paralinguistics Challenge (ComParE) (Schuller et al., 2014) established a unified test-bed for the automatic recognition of speakers' cognitive load in speech, also known as the Cognitive Load Sub-Challenge. It provided transparent conditions

Table 6.2 Systems and results of COMPARE 2014 (Cognitive Load Sub-Challenge)

Reference	Features	Classifier	UAR [%]
Gosztolya et al. (2014)	COMPARE feature set	DNN	63.05
Huckvale (2014)	COMPARE feature set	SVM	63.10
Kua et al. (2014)	COMPARE feature set, MFCCs with shifted delta coefficients, spectral centroid features+Δ+ΔΔ	SVM	63.70
Montacié and Caraty (2014)	COMPARE feature set, IS10 feature set, tempo, pauses, speech events	SVM	63.10
Nwe et al. (2014)	COMPARE feature set, GMM and ANN features	SVM	61.50
Schuller et al. (2014)	COMPARE feature set	SVM	61.60
Van Segbroeck et al. (2014)	IS11 feature set, Perceptual Linear Prediction coefficients, voice-related and Gabor features	i-vector	68.90

Abbreviations: UAR (unweighted average recall)

concerning datasets, toolkits, experiments, and results to the participants. The CLSE database (Section 3.1) was used for evaluation purposes; the official measure was the unweighted average recall (UAR). The systems and results, including the baseline (cf. Schuller et al., 2014), are shown in Table 6.2. Note that one of the participants is not listed in the table, because test results ('slightly worse than the baseline') are not reported in the corresponding paper (see Jing et al., 2014). Almost all participants used the COMPARE baseline feature set (Weninger et al., 2013), which contains 6,373 static features (functionals of low-level feature contours). The baseline features as well as the IS10 and IS11 feature sets are provided by the open-source openSMILE feature extractor (Eyben et al., 2013a). Some works extended these sets by further features (cf. Kua et al., 2014; Montacié and Caraty, 2014; Nwe et al., 2014; Van Segbroeck et al., 2014). SVM is the most frequently encountered classification method (cf. Huckvale, 2014; Kua et al., 2014; Montacié and Caraty, 2014; Nwe et al., 2014; Schuller et al., 2014). In contrast, Gosztolya et al. (2014) used a deep neural network (DNN) architecture. The best result on the test set (68.90% UAR) was achieved with i-vector modelling (cf. Van Segbroeck et al., 2014)—a concept, which was originally proposed by Dehak et al. (2011).

6.2 Experimental Methodology

This section describes the general methodology used in the experiments of this thesis. Figure 6.5 illustrates a unified overview of a single experimental run including model training and evaluation for a given speech database. In the case of k-fold cross-validation, this scheme is repeated k times, with each of the k folds used exactly once as test data and the remaining $k-1$ folds as training data.

At the beginning of the workflow, a given speech database is partitioned into a training set for model construction and a test set for model evaluation. Features are extracted from each instance of both datasets and the corresponding instance labels (target attributes) are assigned to the feature vectors. In order to ensure that the training data is balanced in terms of the class distribution, resampling techniques (Section 6.1.1) can be applied before features are processed further. Since the test data is considered the unseen, independent dataset, it has to be unaffected by this stage. The next stage is feature normalisation (Section 6.1.2). Some methods normalise the training and test partitions individually. Alternatively, the normalisation parameters computed on the training data can be applied to the test data. The opposite direction is not allowed since the test data must remain undiscovered for a fair evaluation. Once the features are normalised, feature selection methods (Section 6.1.3) can be applied to the training data in order to obtain the most relevant features. Afterwards, feature mapping is required, because features obtained from the training data have to exactly match those of the test data. For this purpose, those features of the test data are removed that are not associated with the training data.

Then, model training is performed (Section 6.1.4), which involves providing a learning algorithm with the training data to learn from. The training data contains

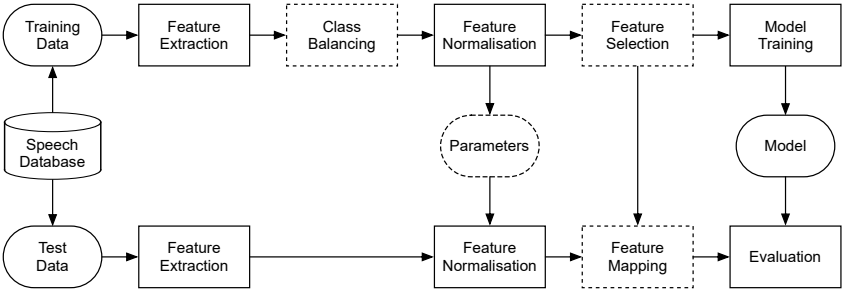


Figure 6.5 A unified overview of the methodology used for the experiments in this thesis. Boxes with rounded corners represent data and models, whereas boxes without rounded corners indicate processing components. Boxes with dashed lines are optional

the labels to map the input data to the target. After the training process, the learned model is obtained. Finally, the model is applied to the test data where it attempts to predict a label for each unseen test instance. Afterwards, the test data is associated with two types of labels, namely the actual labels from the original dataset and the predicted labels produced by the model. Both references form the basis for the evaluation of the model goodness expressed by the measure of interest (Section 6.1.5). If cross-validation is used, results from the folds are averaged to produce a single estimate.

6.3 Within-Corpus Evaluation

In this section, the suitability of various speech features is evaluated for task-specific cognitive load classification. Since all speech corpora used in this thesis (Chapter 3) employ a different methodology for inducing cognitive load, model training and testing are performed for each corpus individually (hereinafter denoted as 'within-corpus evaluation').

This evaluation is accompanied by the question 'Which system configuration performs best concerning a particular task?' and, secondly 'Which features are suitable for a given task?'. The answer to the former question implies the identification of an outstanding classification model for each corpus, whereas the latter refers to an approximate estimate of relevant features for each corpus. In the following sections, the experimental setup and results of the within-corpus evaluation are reported.

6.3.1 Experimental Setup

Speech features were systematically evaluated on the three corpora contained in the CLSE database (Section 3.1) and on the CoLoSS corpus (Section 3.2) for which discretised labels were used. The goal was to automatically discriminate between three levels of cognitive load, i.e., low, medium, and high.

The methodology for determining relevant features in the present investigation can be regarded as a bottom-up approach based on the CL-Extended feature set (Section 5.1.1). It starts with each feature type individually including delta and acceleration coefficients. This is followed by early (feature-level) fusion for each feature group reflecting the different aspects regarding speech characteristics, namely prosody, spectrum, voice quality, and non-linearities in speech production (TEO). Finally, hand-crafted feature sets (CL-Extended, CL-Base, and CL-Stress), introduced in Section 5.1, are examined.

Moreover, late fusion methods regarding the n-best feature group models were of interest, with $n = 2, 3, 4$. That is, the best performing feature groups, each represented by the best feature group result (i.e., single feature type or early fusion), were combined by applying a particular rule for fusing the outputs of the corresponding models. In this context, majority vote and average probability were investigated; WEKA 3 (Witten et al., 2016) was used for this purpose. Regarding majority vote—as the name indicates—the label which was predicted most frequently will be selected as the final output. If labels of a particular classifier have the same probability, then all labels receive a vote. However, one drawback with this rule is that labels are randomly chosen if multiple labels end up with the same number of votes. In the case of the average probability rule, the probabilities of classes given by each model are first summed up and, afterwards, divided by the total number of models.

Apart from knowledge-based approaches, the automatic selection of relevant features was investigated. The following three filter methods were applied to the CL-Extended feature set: the top 100 ranked features by information gain (IG100) as well as correlation (CC_P100) and the correlation-based feature selection (CFS) algorithm (Section 6.1.3.2).

For all experiments, a leave-one-subject-out cross-validation (LOSOCV) was chosen as an evaluation strategy to realise speaker-independent conditions and to have a reasonable approach with regard to the rather small datasets. Since subject information is not included in the test sets of the CLSE database, only the training and development sets were used by combining them in the experiments. Further, it was ensured that each training partition in LOSOCV is balanced in terms of the class distribution. This was achieved by random over-sampling using WEKA 3 (Witten et al., 2016). For CLSE-Span, the required target-size for an approximate balance was 150%. In the case of CoLoSS, the defined target size was 230% to achieve a uniform class distribution. For the remaining corpora, class balancing was not needed. Additionally, different feature normalisation methods were investigated: speaker normalisation, partition normalisation, and training normalisation (Section 6.1.2.2). In order to have fair evaluation conditions, feature normalisation was applied only within the LOSOCV procedure, not to an entire dataset before LOSOCV is performed. Even though subjects of the CLSE database participated across different tasks, each CLSE corpus was considered to be an independent scenario, i.e., speaker normalisation was applied for each corpus individually.

For a consistent evaluation and to investigate the effectiveness of features, only one classification method was used that should generalise well. The chosen classifier was a support vector machine (SVM) with a polynomial kernel function. As training algorithm, Sequential Minimal Optimisation (SMO) (Keerthi et al., 2001) was used, which is implemented in the WEKA 3 data mining toolkit

(Witten et al., 2016). For optimisation, different values of the SVM complexity parameter were investigated with $C = 10^n$, where $n = -5, -4, \ldots, 0$. It can be assumed that higher values would lead to a loss in generalisation properties.

The chosen evaluation measure was the unweighted average recall (UAR), since it provides a truthful estimation of the performance, in particular for test cases in which the distribution of instances is highly unbalanced among classes. The reported results were obtained by averaging UAR values across all folds of LOSOCV. In order to determine whether a system performs significantly better, one-sided paired t-tests were used with $\alpha = 0.05$. That is, two system are compared by considering the differences between (paired) UAR results produced on each fold of LOSOCV.

To summarise, LOSOCV results were produced for 13 feature types, four feature groups (early fusion), three feature sets, and three feature selection methods. Four different corpora, three feature normalisation methods, and six SVM complexities were investigated. Moreover, six late fusion methods were used for each corpus. In total, 1,680 results were obtained.

6.3.2 Results

A series of classification experiments was conducted on four different corpora. Results are reported for optimised SVM complexities; for details, see Appendix B.1.

Recognition results per feature type obtained on CLSE-Time are shown in Table 6.3. Generally, UAR results indicate the issue of speaker variability; speaker normalisation (SN) is generally superior to partition normalisation (PN) and training normalisation (TN). The best prosodic feature type refers to the duration-based features (73.46% UAR) for which SN was used. Early fusion of all speaker-normalised prosodic features leads to a significant increase ($p < 0.05$)—a plus of 6.79%—and to the overall best result for CLSE-Time. Regarding the spectral feature types, formants in conjunction with SN are superior with 67.90% UAR. By uniting all speaker-normalised spectral features, an improvement of 3.70% is achieved, though not significant ($p > 0.05$). In the case of voice quality feature types, the speaker-normalised cepstral peak prominence (CPP) performs best with 66.05% UAR, but there is a drop in performance when fusing CPP with the remaining speaker-normalised voice quality features. Interestingly, the best results in terms of single feature type comparison across all groups are given by Teager energy operator (TEO) based features: 74.07% UAR by NTD-LFPC with SN as well as NFD-LFPC with SN or PN. Little improvement is found by uniting all TEO features (2.47%).

Table 6.4 shows the results per feature type for CLSE-Dual. Looking at the prosodic features, duration yields the best UAR across all normalisation

Table 6.3 Classification results per feature type and normalisation method for CLSE-Time

		UAR [%]		
Feature group	Feature type	SN	PN	TN
Prosodic	Intensity+Δ+ΔΔ	67.28	66.05	56.79
Prosodic	F_0+Δ+ΔΔ	66.67	68.52	54.32
Prosodic	Duration	73.46	70.99	69.75
Prosodic	Early fusion	**80.25**	75.93	70.37
Spectral	Spectral centroid+Δ+ΔΔ	65.43	62.35	51.85
Spectral	12 MFCC+Δ+ΔΔ	64.20	64.81	52.47
Spectral	Formants $F_{1-3}(f,b)$+Δ+ΔΔ	67.90	64.20	51.85
Spectral	Early fusion	**71.60**	69.14	58.64
Voice quality	Jitter+Δ+ΔΔ	53.70	52.47	53.70
Voice quality	Shimmer+Δ+ΔΔ	50.00	50.62	50.00
Voice quality	HNR+Δ+ΔΔ	54.32	46.91	47.53
Voice quality	CPP+Δ+ΔΔ	**66.05**	62.35	51.23
Voice quality	Early fusion	63.58	61.73	57.41
TEO	16 TEO-CB-Auto-Env+Δ+ΔΔ	59.88	62.96	54.94
TEO	12 NTD-LFPC+Δ+ΔΔ	74.07	67.90	54.32
TEO	12 NFD-LFPC+Δ+ΔΔ	74.07	74.07	58.02
TEO	Early fusion	**76.54**	**76.54**	61.73

Best result per feature group is highlighted in bold. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

methods. Unsurprisingly, speaker normalisation is superior with 59.88% UAR. By fusing all speaker-normalised prosodic features, the result is significantly better with an absolute difference of 11.11%. Moreover, with 70.99% UAR, the prosodic (early fusion) system achieves the overall best result for CLSE-Dual. Regarding the group of spectral features, MFCCs in conjunction with SN (61.73% UAR) perform clearly better than the remaining feature types. A slight, but not significant ($p > 0.05$), increase is observed when fusing all spectral features that are normalised by either SN or PN. The performance of voice quality features is generally lower compared to the remaining feature groups; the best result (53.70% UAR) is obtained by CPP with the PN method. Feature type comparison within the TEO group reveals that NTD-LFPC with PN is superior (62.35% UAR). Fusing all TEO features leads to a better performance (65.43% UAR) once the SN method is applied, although the difference is not significant ($p > 0.05$).

Table 6.4 Classification results per feature type and normalisation method for CLSE-Dual

		UAR [%]		
Feature group	Feature type	SN	PN	TN
Prosodic	Intensity+Δ+ΔΔ	54.32	50.62	48.77
Prosodic	F_0+Δ+ΔΔ	40.12	37.04	38.27
Prosodic	Duration	59.88	57.41	50.62
Prosodic	Early fusion	**70.99**	61.73	54.32
Spectral	Spectral centroid+Δ+ΔΔ	51.85	50.00	44.44
Spectral	12 MFCC+Δ+ΔΔ	61.73	54.94	45.68
Spectral	Formants $F_{1-3}(f,b)$+Δ+ΔΔ	49.38	51.23	44.44
Spectral	Early fusion	**64.81**	62.35	52.47
Voice quality	Jitter+Δ+ΔΔ	45.68	39.51	40.74
Voice quality	Shimmer+Δ+ΔΔ	43.83	41.36	44.44
Voice quality	HNR+Δ+ΔΔ	45.68	42.59	43.21
Voice quality	CPP+Δ+ΔΔ	48.77	**53.70**	47.53
Voice quality	Early fusion	50.62	47.53	43.21
TEO	16 TEO-CB-Auto-Env+Δ+ΔΔ	44.44	43.21	38.27
TEO	12 NTD-LFPC+Δ+ΔΔ	61.73	62.35	45.68
TEO	12 NFD-LFPC+Δ+ΔΔ	61.11	61.11	48.77
TEO	Early fusion	**65.43**	61.73	51.85

Best result per feature group is highlighted in bold. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

For CLSE-Span, recognition results per feature type are shown in Table 6.5. The performance is generally lower compared to CLSE-Time and CLSE-Dual. It can be seen that the overall best result for CLSE-Span with 54.74% UAR is obtained by the speaker-normalised duration-based features. Even if early fusion is used for the prosodic features, there is no improvement in performance. Within the group of spectral features, MFCCs with SN perform best (48.77% UAR). An insignificant improvement is observed when fusing all speaker-normalised spectral features (49.79% UAR), which corresponds to the second best result for CLSE-Span. Regarding voice quality features, the CPP with SN is superior (40.62% UAR), but a decrease in performance can be seen when CPP is combined with the remaining speaker-normalised voice quality features. However, fusing all features of this group has a positive effect in the case of PN and TN. For the group of TEO-based features, NFD-LFPC outperforms the remaining TEO

Table 6.5 Classification results per feature type and normalisation method for CLSE-Span

		UAR [%]		
Feature group	Feature type	SN	PN	TN
Prosodic	Intensity+Δ+ΔΔ	46.02	43.71	44.30
Prosodic	F_0+Δ+ΔΔ	45.17	40.31	39.21
Prosodic	Duration	**54.74**	53.66	52.82
Prosodic	Early fusion	52.73	50.21	49.97
Spectral	Spectral centroid+Δ+ΔΔ	39.83	42.14	40.25
Spectral	12 MFCC+Δ+ΔΔ	48.77	48.30	43.99
Spectral	Formants $F_{1-3}(f,b)$+Δ+ΔΔ	45.70	45.93	44.48
Spectral	Early fusion	**49.79**	48.98	46.74
Voice quality	Jitter+Δ+ΔΔ	37.07	38.89	38.09
Voice quality	Shimmer+Δ+ΔΔ	36.86	36.73	37.16
Voice quality	HNR+Δ+ΔΔ	37.33	36.77	37.85
Voice quality	CPP+Δ+ΔΔ	**40.62**	38.68	39.22
Voice quality	Early fusion	39.64	40.38	40.27
TEO	16 TEO-CB-Auto-Env+Δ+ΔΔ	42.70	42.52	43.31
TEO	12 NTD-LFPC+Δ+ΔΔ	47.90	47.85	45.29
TEO	12 NFD-LFPC+Δ+ΔΔ	**49.46**	49.32	47.73
TEO	Early fusion	49.13	48.42	46.52

Best result per feature group is highlighted in bold. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

systems across all feature normalisation methods—early fusion of TEO features stays behind. Unsurprisingly, the best result is obtained by using SN (49.46% UAR).

Next, the results per feature type for CoLoSS are examined. From Table 6.6 it can be seen that, in contrast to the other corpora where SN dominates (cf. Table 6.3 to 6.5), the TN method is superior for CoLoSS across all features. This difference can be explained by the fact that the data of CoLoSS is highly unbalanced regarding the class distribution—applying feature normalisation after class balancing (cf. Section 6.2) is probably accompanied by the risk that it causes a mismatch between training and test data. This mismatch seems to be compensated by TN, which normalises the training and test partition with parameters computed only from the training partition; this topic deserves more attention in future work. Looking at the results, the most effective features for

Table 6.6 Classification results per feature type and normalisation method for CoLoSS

		UAR [%]		
Feature group	Feature type	SN	PN	TN
Prosodic	Intensity+Δ+ΔΔ	41.91	46.07	58.13
Prosodic	F_0+Δ+ΔΔ	39.11	41.25	53.66
Prosodic	Duration	44.93	56.06	**62.67**
Prosodic	Early fusion	45.75	56.41	61.41
Spectral	Spectral centroid+Δ+ΔΔ	39.59	41.60	50.92
Spectral	12 MFCC+Δ+ΔΔ	43.24	50.33	54.92
Spectral	Formants $F_{1-3}(f,b)$+Δ+ΔΔ	41.27	49.88	52.80
Spectral	Early fusion	43.03	52.34	**54.99**
Voice quality	Jitter+Δ+ΔΔ	37.92	42.53	49.50
Voice quality	Shimmer+Δ+ΔΔ	36.47	42.11	47.27
Voice quality	HNR+Δ+ΔΔ	36.11	37.52	47.34
Voice quality	CPP+Δ+ΔΔ	39.93	43.84	**53.77**
Voice quality	Early fusion	35.94	45.69	52.59
TEO	16 TEO-CB-Auto-Env+Δ+ΔΔ	38.78	47.05	52.28
TEO	12 NTD-LFPC+Δ+ΔΔ	50.83	50.30	57.47
TEO	12 NFD-LFPC+Δ+ΔΔ	50.56	50.82	**57.54**
TEO	Early fusion	46.73	50.36	55.44

Best result per feature group is highlighted in bold. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

CoLoSS refer to the prosody, in particular to the duration with 62.67% UAR. MFCCs are superior in terms of single spectral feature types (54.92% UAR). A slight, but not significant, increase with an absolute difference of 0.07% is observed when uniting all spectral features. The best voice quality feature type is the CPP (53.77% UAR)—it performs even better than early fusion of all voice quality features. Regarding the TEO group, the highest UAR with 57.54% is obtained by NFD-LFPC; the performance is comparable with that of NTD-LFPC (the absolute difference is 0.07%). Fusing all TEO-based features results in lower performance (55.44% UAR).

So far, the performance of feature types and their fusion per feature group were investigated. The *n*-best feature group models, each represented by the best feature group result, were afterwards combined by late fusion methods. Table 6.7 shows the results for all four corpora. For comparison purposes, the overall best feature group result per corpus is also shown; it is denoted as '1-best' in the table (cf. Table 6.3 to 6.6). Regarding CLSE-Time, fusing the outputs of the 3-best

Table 6.7 Classification results per late fusion method for all four corpora

		UAR [%]			
n-best	Late fusion	CLSE-Time	CLSE-Dual	CLSE-Span	CoLoSS
1-best	—	80.25	**70.99**	54.74	**62.67**
2-best	Majority vote	79.01	67.28	53.57	59.88
2-best	Average probability	79.01	67.28	53.57	59.88
3-best	Majority vote	80.25	69.75	**54.92**	60.26
3-best	Average probability	**80.86**	69.75	54.80	58.92
4-best	Majority vote	77.78	65.43	52.84	58.43
4-best	Average probability	78.40	66.05	53.04	57.83

Best result per corpus is highlighted in bold. Abbreviations: UAR (unweighted average recall)

models by average probability outperforms—not significantly ($p > 0.05$)—the 1-best model with 80.86% UAR. An improvement is also observed for CLSE-Span. By applying majority vote for the 3-best models, up to 54.92% UAR is reached; the improvement is not significant at a level of 5%. For the remaining corpora, late fusion methods perform worse than the 1-best models.

For the feature set evaluation, the full feature set (CL-Extended) is compared to knowledge-based subsets (CL-Base and CL-Stress). The classification results for all four corpora are shown in Table 6.8. For CLSE-Time, the best result is obtained using CL-Base with the SN method (81.48% UAR). It is, however, not significantly better than that of CL-Extended (with PN). Regarding CLSE-Dual, CL-Base with speaker normalisation also leads to the highest UAR (72.22%), though the difference to CL-Extended (with SN) is not significant. In the case of CLSE-Span, CL-Extended with PN (54.42% UAR) outperforms the remaining configurations (the difference is not significant). For CoLoSS, the TN method yields by far the best results, which is not surprising because TN was also the optimal feature normalisation method in the feature type evaluation (cf. Table 6.6). The results show that CL-Stress is slightly better than CL-Extended, but CL-Base is (not significantly) superior with 62.02% UAR.

Next, filter-based feature selection methods are investigated. The results are summarised in Table 6.9. The classification performances are compared to the full feature set (CL-Extended); the best system configuration in terms of the full feature set serves as a baseline. Looking at the results, no improvements can be observed for CLSE-Time as well as CoLoSS. Regarding CLSE-Dual, the best performance (72.84% UAR) is achieved by CFS for speaker-normalised features, although it is not significantly better than the full set ($p > 0.05$). Interestingly, when considering only TN for CLSE-Dual, all feature selection methods outperform the full set. Notable results are obtained on CLSE-Span; for each

Table 6.8 Classification results per feature set and normalisation method for all four corpora

		UAR [%]		
Corpus	Feature set	SN	PN	TN
CLSE-Time	CL-Extended	79.01	80.86	67.28
CLSE-Time	CL-Base	**81.48**	78.40	72.22
CLSE-Time	CL-Stress	79.01	79.01	63.58
CLSE-Dual	CL-Extended	70.99	69.14	54.94
CLSE-Dual	CL-Base	**72.22**	64.81	54.32
CLSE-Dual	CL-Stress	65.43	59.26	54.32
CLSE-Span	CL-Extended	53.50	**54.42**	51.15
CLSE-Span	CL-Base	53.05	52.27	49.09
CLSE-Span	CL-Stress	52.89	51.75	50.18
CoLoSS	CL-Extended	47.23	54.78	59.15
CoLoSS	CL-Base	44.85	58.06	**62.02**
CoLoSS	CL-Stress	46.54	56.62	60.51

Best result per corpus is highlighted in bold. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

normalisation method, feature selection yields better results than the full set. The best result for CLSE-Span (57.37% UAR) is obtained by CC_P100 when using speaker normalisation, which corresponds to an increase of 2.95% (not significant) compared with the best full set configuration.

In the following, optimal system settings are discussed. The best configuration per corpus is given in Table 6.10. It can be seen that only feature subsets of the full set (CL-Extended) are superior. Some of them are hand-crafted, while others are obtained automatically by using filter-based feature selection. Large differences are observed between UAR results. The absolute difference between the highest UAR (CLSE-Time) and the lowest UAR (CLS-Span) is 24.11%. Generally, CLSE-Span and CoLoSS seem to be more challenging than the variants of the Stroop test.

All four corpora contain a relatively small number of instances recorded under laboratory conditions. This raises questions about the applicability of the proposed systems to real-world use-cases. Hence, an estimate of the optimal system configuration should be made for each task instead of suggesting outstanding models. For this purpose, the top three systems per task are considered, whereby configurations with equivalent results are regarded as one of the best systems—meaning that more than three configurations can be given for a particular corpus.

Table 6.9 Classification results per feature selection method and normalisation method for all four corpora. The full set (CL-Extended) is also shown for comparison purposes

		UAR [%]		
Corpus	Feature selection	SN	PN	TN
CLSE-Time	Full set (CL-Extended)	79.01	**80.86**	67.28
CLSE-Time	IG100	69.14	77.16	77.16
CLSE-Time	CC_P100	72.84	74.07	74.07
CLSE-Time	CFS	75.93	79.01	79.01
CLSE-Dual	Full set (CL-Extended)	70.99	69.14	54.94
CLSE-Dual	IG100	69.75	67.28	67.28
CLSE-Dual	CC_P100	69.75	69.14	69.14
CLSE-Dual	CFS	**72.84**	68.52	68.52
CLSE-Span	Full set (CL-Extended)	53.50	54.42	51.15
CLSE-Span	IG100	55.69	54.58	54.58
CLSE-Span	CC_P100	**57.37**	56.43	56.43
CLSE-Span	CFS	56.76	55.77	55.77
CoLoSS	Full set (CL-Extended)	47.23	54.78	**59.15**
CoLoSS	IG100	48.79	46.26	46.26
CoLoSS	CC_P100	47.52	57.21	57.21
CoLoSS	CFS	44.77	53.93	53.93

Best result per corpus is highlighted in bold. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

Table 6.10 Best system configuration for each corpus

Corpus	Features	Norm	C	UAR [%]
CLSE-Time	Feature set: CL-Base	SN	10^{-2}	81.48
CLSE-Dual	Feature selection: CFS	SN	10^{-2}	72.84
CLSE-Span	Feature selection: CC_P100	SN	10^{-3}	57.37
CoLoSS	Feature type: Duration	TN	10^{-2}	62.67

Abbreviations: C (SVM complexity parameter), Norm (feature normalisation method), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

The systems are summarised in Table 6.11. Shown are the feature groups (single feature type or feature fusion), proposed feature sets, and feature selection methods. From the table it can be seen that the greatest share refers to both the group of prosodic features and the CL-Base feature set. However, CL-Base should be preferred, since it covers more aspects in speech, and secondly, one of the

Table 6.11 Summary of the top three system configurations for each corpus

Configuration	Corpora				$\sum$
	CLSE-Time	CLSE-Dual	CLSE-Span	CoLoSS	
Features					
Feature group: Prosodic	✓*	✓		✓	3
Feature group: Spectral	✓*				1
Feature group: Voice quality					0
Feature group: TEO	✓*				1
Feature set: CL-Extended	✓	✓			2
Feature set: CL-Base	✓	✓		✓	3
Feature set: CL-Stress					0
Feature selection: IG100					0
Feature selection: CC_P100			✓		1
Feature selection: CFS		✓	✓		2
Feature normalisation					
Speaker normalisation (SN)	✓	✓	✓		3
Partition normalisation (PN)	✓		✓		2
Training normalisation (TN)			✓	✓	2

*: feature group model for late fusion (cf. Table 6.3 and 6.7)

prosodic systems is only used for late fusion (CLSE-Time, cf. Table 6.3 and 6.7), not as a final system. It is notable that the best results in the case of CLSE-Span are obtained exclusively by feature selection methods. The most common feature selection method across tasks is CFS, which is generally an advisable method because it considers both feature relevance and feature redundancy. Regarding the feature normalisation methods, speaker normalisation should be preferred, if possible. On the other hand, there are systems regarding CLSE-Span that perform well when using PN and TN (cf. Table 6.9). The CoLoSS corpus serves as an example where the best results are obtained only by TN (cf. Table 6.6, 6.8, and 6.9). In this context, it is worth noting that the choice of a feature normalisation method strongly depends on the nature of the dataset (not only on the task!), as discussed above.

6.4 Cross-Corpus Evaluation

The goal of this experiment is to explore the transferability of a cognitive load classification system to similar domains. Experiments are usually conducted using a single database where the training and test set belong to the same context.

Testing on different speech-based cognitive load corpora requires systems that generalise across different tasks, different types of labels, different languages, and different recording conditions. In previous works, it was assumed that an impression of the generalisation ability of a recognition system can be obtained by simple cross-corpus evaluation (e.g., Schuller et al., 2010, 2011b). This approach was followed in this thesis and is associated to the following two questions: (1) 'What is the strength of generalisability of a model, which is trained on a single corpus?'; (2) 'Can the performance be improved by combining several corpora for model training?'. Since this evaluation is concerned with aspects of generalisation, both questions are accompanied by the issue of optimal generic feature sets.

There exists only a limited number of cognitive load corpora, while the methods applied for inducing cognitive load are rather different. This poses the question of whether there is any way to generalise a system across different cognitive load tasks. In contrast, an increasing amount of data is available for speech emotion recognition. This provided the basis for many studies to investigate the generalisability and robustness of systems by conducting cross-corpus experiments using a variety of speech data, for instance, from acted, induced, and natural emotions (see Lefter et al., 2010; Schuller et al., 2010, 2011b,c).

In the following sections, the experimental setup and the results of the cross-corpus evaluation are presented, with the focus on the above-mentioned research questions.

6.4.1 Experimental Setup

Cognitive load recognition was considered a ternary classification problem (low, medium, and high load). The experiments were conducted using the three corpora of the CLSE database (Section 3.1) and the CoLoSS corpus (Section 3.2). For the CoLoSS corpus, the discretised labels were employed. In the first part of the experiments, each model is trained on the basis of a single corpus, which is then tested on the data of the remaining corpora. The second part of the experiments addresses corpus fusion by the agglomeration of various corpora on instance-level—also known as data pooling—for the construction of a model, which is tested on disjoint data from a separate corpus.

For generalisation purposes, only hand-crafted feature sets were used instead of exploring feature types or feature selection methods. These include CL-Extended, CL-Base, and CL-Stress (Section 5.1).

All experiments are speaker-independent. In the case of the CLSE corpora, only the training and development sets were used by combining them, because subject IDs are not included in the corresponding test sets. Since all subjects

of the CLSE database participated in all three CLSE tasks, a kind of leave-one-subject-out validation was required if both training and test corpora are part of the CLSE database. For instance, if CLSE-Time is part of the training data and CLSE-Dual is the corpus to be tested, then the kth fold of the validation represents subject k in CLSE-Dual for testing while k is excluded from CLSE-Time for training. Moreover, it was ensured that each corpus contained in the training data is balanced in terms of the distribution among classes. This was achieved by applying WEKA's implementation of random over-sampling (Witten et al., 2016). For CLSE-Span and CoLoSS, the required target-size was 150% and 230%, respectively. For the remaining corpora, class balancing was not required. Further, the following feature normalisation methods were investigated: speaker normalisation, corpus normalisation, and training normalisation (Section 6.1.2.2). Although one can obtain more data for speaker normalisation by taking information on subjects across the CLSE corpora into account, per speaker normalisation was performed on each corpus individually to ensure task independence. While corpus normalisation adjusts feature values on a per corpus basis for both the training and test set, the training normalisation method ignores corpora of the training set and applies the computed normalisation parameters to the test set.

Classification was carried out by a support vector machine (SVM) classifier with a polynomial kernel; Sequential Minimal Optimisation (SMO) (Keerthi et al., 2001) was used as training algorithm, which is provided by the WEKA 3 toolkit (Witten et al., 2016). In the experiments, the SVM complexity parameter C was optimised by investigating $C = 10^n$, with $n = -5, -4, \ldots, 0$. Higher values were avoided for generalisation purposes.

The unweighted average recall (UAR) was chosen as the evaluation measure because the distribution among classes is highly unbalanced in the case of CLSE-Span and CoLoSS. If the leave-one-subject-out procedure (see above) was applied, UAR results from the folds were averaged to produce a single estimate.

In summary, the first part of the experiments yields 648 results: Four corpora were used for single-corpus training, whereby each model is tested against the remaining three corpora; three feature sets, three normalisation methods, and six SVM complexities were systematically evaluated. In the second part of the experiments, 216 results were obtained: Multi-corpus training was performed using three out of four corpora, whereby the resulting model was tested against the fourth one. This gives four test cases, each for three feature sets, three feature normalisation methods, and six SVM complexities.

6.4.2 Results

Classification results for cross-corpus evaluation were produced by two types of experiments: (1) single-corpus training, whereby the resulting model is tested

against the remaining corpora; (2) combining various corpora by data pooling for training and using a separate corpus for testing. The results are reported in terms of optimal support vector machine (SVM) *C* values. For reproducibility, the reader is referred to Section 6.4.1 and Appendix B.2.

First, cross-corpus evaluation by single-corpus training is considered. The results per feature set and feature normalisation method are found in Table 6.12. The UAR values were obtained by averaging single UAR results produced on each of the test corpora with optimal constant SVM *C*. As expected—due to divergent task characteristics—the results are far worse compared to those of the within-corpus evaluation (cf. Section 6.3). It can be seen that CL-Base yields highest UAR for each training set. There are notable differences with regard to feature normalisation: while speaker normalisation (SN) is superior for CLSE-Dual and CoLoSS, corpus normalisation (CN) and training normalisation (TN) are optimal for CLSE-Time and CLSE-Span, respectively. The overall best result is achieved by CLSE-Time with the corpus normalisation method (41.27% UAR). However, the results are only little above the chance level (33.33%) and obviously too low for practical use.

Table 6.12 Classification results for cross-corpus evaluation by training on one corpus and testing against the remaining three corpora with constant SVM *C* parameter. Comparison of feature sets and normalisation methods

		UAR [%]		
Train on	Feature set	SN	CN	TN
CLSE-Time	CL-Extended	38.53	37.47	35.00
CLSE-Time	CL-Base	38.69	**41.27**	36.22
CLSE-Time	CL-Stress	39.31	39.20	35.04
CLSE-Dual	CL-Extended	38.60	36.58	34.53
CLSE-Dual	CL-Base	**40.33**	37.43	35.34
CLSE-Dual	CL-Stress	39.20	38.98	35.97
CLSE-Span	CL-Extended	32.23	34.86	33.75
CLSE-Span	CL-Base	33.29	33.95	**36.19**
CLSE-Span	CL-Stress	33.37	33.36	33.74
CoLoSS	CL-Extended	39.43	35.98	33.61
CoLoSS	CL-Base	**40.48**	35.36	33.61
CoLoSS	CL-Stress	39.48	37.12	33.63

Best result per corpus is highlighted in bold. Abbreviations: CN (corpus normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

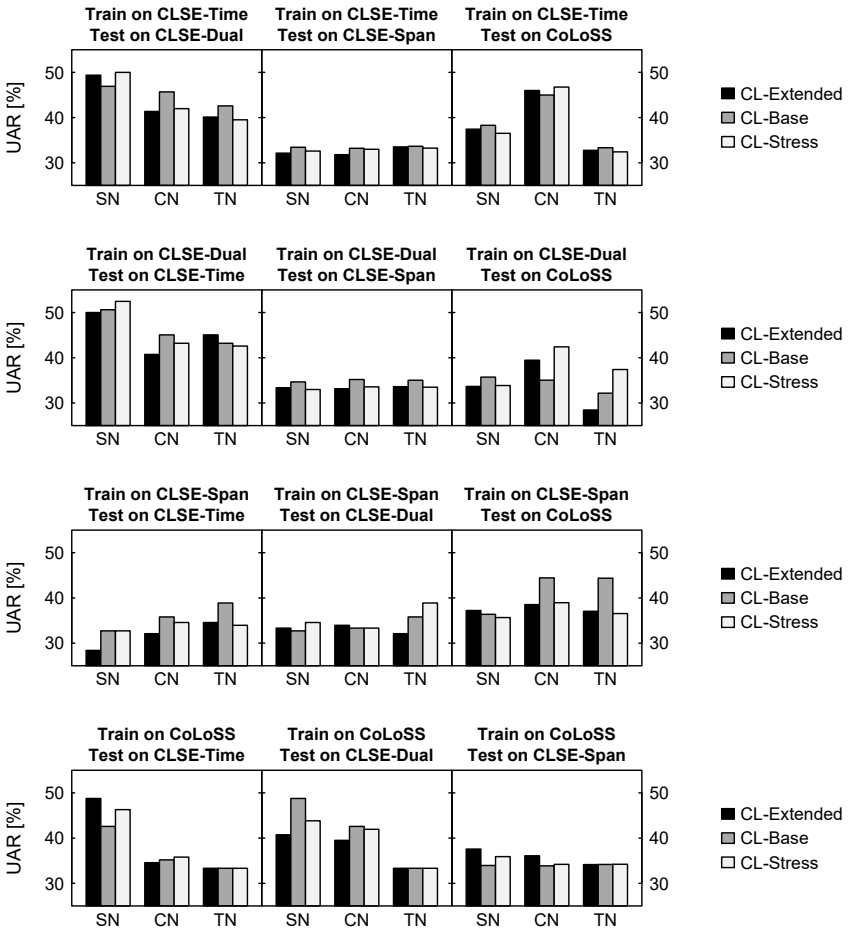


Figure 6.6 Classification results for cross-corpus evaluation by training on one corpus and testing against the remaining corpora with optimised SVM complexity. Comparison of feature sets (CL-Extended, CL-Base, and CL-Stress) and feature normalisation methods (SN: speaker normalisation, CN: corpus normalisation, and TN: training normalisation)

Additional UAR results are depicted in Figure 6.6. Here, the SVM complexity parameter C is optimised for each test case individually. The figure indicates that the classification models are not capable of generalising well across all test corpora. However, individual cases show results of around 50% UAR. For instance, when using CLSE-Time as the training set in connection with speaker-

normalised CL-Stress features, the best result is obtained on CLSE-Dual (50.00% UAR) and vice versa (52.47% UAR). Nearly identical results are given for CLSE-Time when testing on the CoLoSS corpus. In turn, models trained on CoLoSS reach almost 50% UAR when they are applied to CLSE-Time or CLSE-Dual. The data of CLSE-Span is more challenging—test results are barely above the chance level. When using CLSE-Span for model training, results are generally very low (the best is 44.44% UAR by using CL-Base with CN, tested on CoLoSS). Finally, a general recommendation regarding the feature sets and normalisation methods can hardly be derived, since optimal configurations very much depend on individual cases.

The results of the second cross-corpus experiment are summarised in Table 6.13. Each result was produced by using one out of four corpora for testing, while data of the remaining corpora were pooled on instance-level and used for model training with optimised SVM complexity C. In order to get an impression of the system's performance in general with respect to feature sets and feature normalisation methods, the mean across optimal UAR results is also shown in the table. Moreover, results for leave-one-corpus-out cross-validation (LOCOCV) are reported that were obtained by optimal constant SVM C across test cases. Regarding test results produced on each individual corpus, the highest UAR values are: 45.68% UAR on CLSE-Time (CL-Stress with SN), 43.83% UAR on CLSE-Dual (CL-Base with SN), 36.49% UAR on CLSE-Span (CL-Extended with SN), and 46.34% UAR on CoLoSS (CL-Base with CN). By taking the mean UAR results into consideration, CL-Base outperforms the other feature sets for SN and CN with 39.87% UAR and 38.10% UAR, respectively. The best result for TN (36.41% mean UAR) is achieved by using the CL-Extended feature set. The same applies for LOCOCV, but—as expected—the results are generally lower than those of the mean UAR since SVM C remains constant across the folds in LOCOCV.

In summary, cross-corpus evaluation is a challenging field due to divergent characteristics of corpora. The first part of the cross-corpus evaluation demonstrates model generalisability for cognitive load classification by using single task-specific corpora as training data. In this context, the generalisation capability can be regarded as the highest average UAR score determined by testing across all remaining corpora with a constant SVM complexity C. In light of this criterion, the results are 41.27% for CLSE-Time, 40.33% for CLSE-Dual, 36.19% for CLSE-Span, and 40.48% for CoLoSS (cf. Table 6.12). Taking individual cases into account leads to better results, even a plus of about 10% (cf. Figure 6.6). Hence, it is recommended to apply task-specific cognitive load assessment systems to domains that meet the requirements of the system to some extent. Such is the case, in particular, for CLSE-Time and CLSE-Dual, since both corpora employ the Stroop test in conjunction with stressful conditions. However,

Table 6.13 Classification results for cross-corpus evaluation by corpus fusion for model training and testing against a different, single corpus. Comparison of feature sets and normalisation methods

		UAR [%]		
Test on	Feature set	SN	CN	TN
CLSE-Time	CL-Extended	38.89	32.10	36.42
CLSE-Dual	CL-Extended	38.89	40.74	35.19
CLSE-Span	CL-Extended	36.49	35.39	34.61
CoLoSS	CL-Extended	38.29	40.91	39.42
Mean	CL-Extended	38.14	37.29	36.41
LOCOCV	CL-Extended	36.92	34.17	35.17
CLSE-Time	CL-Base	43.21	33.95	33.33
CLSE-Dual	CL-Base	43.83	38.89	35.80
CLSE-Span	CL-Base	33.66	33.21	33.79
CoLoSS	CL-Base	38.77	46.34	39.31
Mean	CL-Base	**39.87**	38.10	35.56
LOCOCV	CL-Base	**39.84**	37.92	34.74
CLSE-Time	CL-Stress	45.68	33.95	33.33
CLSE-Dual	CL-Stress	41.36	39.51	37.04
CLSE-Span	CL-Stress	34.71	34.09	33.53
CoLoSS	CL-Stress	37.50	40.12	35.58
Mean	CL-Stress	39.81	36.92	34.87
LOCOCV	CL-Stress	38.77	35.45	33.94

Mean UAR refers to the test results shown, each obtained by optimised SVM *C*, while leave-one-corpus-out cross-validation (LOCOCV) results are obtained by using optimal constant SVM *C* across all test cases. Best mean UAR and LOCOCV are highlighted in bold. Abbreviations: CN (corpus normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

the results are far below those of the within-corpus experiments (Section 6.3). Another aspect of cross-corpus evaluation concerns testing across a set of corpora, while for each test case the remaining corpora are fused by data pooling for model training. The results suggest that the CL-Base feature set with speaker normalisation is superior on average for individual cases (39.87% UAR) and for LOCOCV (39.84% UAR). However, with regard to individual test results, data pooling generally yields lower performances compared to single-corpus training (cf. Table 6.13 and Figure 6.6). Consequently, as stated above, individual cases should be considered in which both training and test data exhibit similar characteristics in terms of the cognitive load task.

6.5 Mixed-Corpus Evaluation

While training and test data are disjunctive in terms of corpora in the cross-corpus evaluation (Section 6.4), the mixed-corpus methodology aims at combining a number of corpora for model training, whereby a subset of one out of these corpora is excluded for testing. In this way, various contexts are involved in the modelling process, while at the same time knowledge about the target domain is used. Lefter et al. (2010) already demonstrated positive effects of the mixed-corpus method on the recognition performance in comparison to cross-corpus and within-corpus approaches in the field of speech emotion recognition.

The evaluation presented here refers to the question: 'How much potential for improvement exists through combining data from the target domain and other, similar domains for cognitive load modelling in comparison to within-corpus and cross-corpus approaches?'. In this connection, the performance of hand-crafted feature sets is of interest. The experimental setup and results of the mixed-corpus evaluation are given in the following sections.

6.5.1 Experimental Setup

In the experiments, the speakers' cognitive load state had to be classified automatically as either low, medium, or high. Features and algorithms were evaluated on four different corpora, namely CLSE-Time, CLSE-Dual, and CLSE-Span from the CLSE database (Section 3.1) and the CoLoSS corpus (Section 3.2). Note that the CoLoSS corpus was used together with discretised labels. In order to foster the generalisability of models, the CL-Extended, CL-Base, and CL-Stress feature set (Section 5.1) were employed instead of single feature types or feature selection methods.

Leave-one-subject-out cross-validation (LOSOCV) was performed to have speaker-independent and comparable conditions. Since subject IDs are not included in the test partitions of the CLSE database, only the data of the training and development sets were used by combining them. The mixed-corpus method was realised as follows: A particular corpus is the focus of the evaluation, meaning that the LOSOCV procedure is applied only to this corpus. At this juncture—from a technical point of view—, this corresponds to the strategy used in the within-corpus evaluation (Section 6.3). The difference in the evaluation carried out here is that for each fold of LOSOCV, the remaining three corpora are added to the training partition. For a valid evaluation, it was ensured that the speaker contained in the test partition was not present in any other corpus of the training partition.

The goal was to realise a mixture of corpus data. The (pooled) training partitions of LOSOCV were balanced regarding the class distribution by random

over-sampling using WEKA 3 (Witten et al., 2016). The required target-size for an approximate data balance is 200%. Feature normalisation parameters were computed across corpora, not for each corpus separately. The following three feature normalisation methods were investigated: speaker normalisation, partition normalisation, and training normalisation (Section 6.1.2.2). Regarding per speaker normalisation, the features of a particular speaker were normalised by considering the overall speaker context from all corpora, if possible. In fact, this relates to the corpora of the CLSE database. It should be noted that the overall speaker context also applies to test partitions.

As in the previous evaluations, the WEKA 3 data mining toolkit (Witten et al., 2016) was used for classification, with polynomial kernel support vector machines (SVMs) trained using Sequential Minimal Optimisation (SMO) (Keerthi et al., 2001). The following SVM complexities were investigated: $C = 10^n$, with $n = -5, -4, \ldots, 0$. As evaluation measure, the unweighted average recall (UAR) was chosen due to the unbalanced distributions among classes in the CLSE-Span and CoLoSS datasets.

To summarise, results were produced on four corpora with three feature sets, three feature normalisation methods, and six SVM complexities. This yields a total of 216 results.

6.5.2 Results

In this section, results of the mixed-corpus evaluation are reported. The classification performances, optimised by means of the support vector machine (SVM) complexity (see Appendix B.3), are shown in Table 6.14. Different feature sets (CL-Extended, CL-Base, and CL-Stress) and feature normalisation methods (SN: speaker normalisation, PN: partition normalisation, and TN: training normalisation) are compared.

The best UAR results can be roughly summarised as follows: 50.62% on CLSE-Time (CL-Stress with SN), 46.30% on CLSE-Dual (CL-Base with TN and CL-Stress with PN), 50.51% on CLSE-Span (CL-Extended with SN), and 58.50% on CoLoSS (CL-Extended with TN). Judging from the results, there is no single best feature set across test cases and the differences between results are rather small.

Compared to the performance of feature sets in the within-corpus evaluation (Table 6.8 in Section 6.3.2), no further improvements can be observed, though the difference is small in the case of CLSE-Span and CoLoSS (3.91% and 3.52%, respectively). In contrast, a clear difference can be seen for CLSE-Time (30.86%) and CLSE-Dual (25.92%). Unsurprisingly, the best results obtained by mixed-corpus experiments are slightly better than those obtained by systems for which

Table 6.14 Classification results for mixed-corpus evaluation. Comparison of feature sets and normalisation methods

		UAR [%]		
Test on	Feature set	SN	PN	TN
CLSE-Time	CL-Extended	46.30	44.44	41.98
CLSE-Dual	CL-Extended	45.68	41.98	37.65
CLSE-Span	CL-Extended	50.51	45.08	44.12
CoLoSS	CL-Extended	44.85	47.41	58.50
Mean	CL-Extended	46.84	44.73	45.56
CLSE-Time	CL-Base	48.77	44.44	43.21
CLSE-Dual	CL-Base	39.51	44.44	46.30
CLSE-Span	CL-Base	48.00	42.08	44.06
CoLoSS	CL-Base	45.34	46.26	58.34
Mean	CL-Base	45.41	44.31	47.98
CLSE-Time	CL-Stress	50.62	40.74	45.06
CLSE-Dual	CL-Stress	44.44	46.30	45.06
CLSE-Span	CL-Stress	49.71	43.37	44.28
CoLoSS	CL-Stress	45.78	45.71	58.48
Mean	CL-Stress	47.64	44.03	**48.22**

Best mean UAR is highlighted in bold. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation), UAR (unweighted average recall)

the target domain was not considered for modelling (cf. second experiment of the cross-corpus evaluation, Table 6.13 in Section 6.4.2). More precisely, the absolute differences are 4.94% for CLSE-Time, 2.47% for CLSE-Dual, 14.02% for CLSE-Span, and 12.16% for CoLoSS. Looking at the mean UAR results in Table 6.14, it can be seen that the CL-Stress feature set with the TN method performs best in general (48.22% UAR), which corresponds to a plus of 8.35% compared with the best mean UAR obtained by cross-corpus experiments (see again Table 6.13 in Section 6.4.2).

It can be concluded from the findings of this evaluation that the model performance of the proposed mixed-corpus method lies in between those of within-corpus and cross-corpus experiments. Future research effort could be spent on corpus or instance selection—for example, based on the distance between class centres (cf. Schuller et al., 2011b)—to better exploit the amount of training material available for cognitive load recognition.

6.6 ComParE 2014—Cognitive Load Sub-Challenge

The INTERSPEECH 2014 Computational Paralinguistics Challenge (ComParE) (Schuller et al., 2014) established a unified test-bed for the automatic recognition of speakers' cognitive and physical load in speech. The experiments presented in this section focus on optimisation problems regarding features and algorithms in order to outperform the ComParE baseline system according to the rules of the Cognitive Load Sub-Challenge.

Three levels of cognitive load have to be classified automatically: low (L_1), medium (L_2), and high (L_3). The CLSE database serves to evaluate the performance of systems. The database is partitioned speaker-disjunctive into training, development, and test sets (cf. Section 3.1.2), whereby each partition consists of three different tasks (CLSE-Time, CLSE-Dual, and CLSE-Span). While information on tasks is available for each partition, subject information is not included in the test set. The official evaluation measure of the Cognitive Load Sub-Challenge is the unweighted average recall (UAR) in %.

The baseline system is characterised as follows: The open-source feature extractor openSMILE (2.0 release) (Eyben et al., 2013a) was applied for the extraction of features on utterance-level. The features are provided by the ComParE feature set, which is defined in the corresponding openSMILE configuration file. It contains 6,373 statistical features (functionals of low-level descriptors); for details on the feature set, the reader is referred to Weninger et al. (2013). Balancing the training instances in terms of the class distribution was achieved by applying integer upsampling to the classes L_1, L_2, and L_3 by the factors 2, 2, and 3, respectively. CLSE-Time, CLSE-Dual, and CLSE-Span were treated separately due to the different task characteristics. The baseline system uses z-score transformation regarding feature normalisation for training and test partitions with parameters computed only from the training partition—which corresponds to the training normalisation (TN) method used in this thesis (cf. Section 6.1.2.2). For classification, a linear kernel support vector machine (SVM) provided by the WEKA 3 data mining toolkit (Witten et al., 2016) was chosen and trained using the Sequential Minimal Optimisation (SMO) algorithm (Keerthi et al., 2001). The SVM complexity parameter C was investigated in the development phase. Optimal results were obtained by $C = 10^{-4}$. For the evaluation on the test set, a new model was trained for each task using the corresponding training and development set, whereby the integer upsampling method was applied once again. The development baseline is 63.2% UAR (74.6% on CLSE-Time, 63.5% on CLSE-Dual, and 61.2% on CLSE-Span) and the official baseline on the test set is 61.6% UAR (66.7% on CLSE-Time, 56.9% on CLSE-Dual, and 61.5%

on CLSE-Span). It should be noted that there exists only a single true baseline, namely the test set result obtained by the official baseline system. Further details are found in the challenge paper (Schuller et al., 2014).

As stated above, the goal is to outperform the official baseline system. In this respect, the question of suitable features is of interest. Secondly, since subject information is not included in the test partition, there is the question of whether the performance can be improved by adjusting features on a per speaker basis in a way that different clusters, obtained by a speaker identification technique, are normalised individually.

6.6.1 Experimental Setup

It is assumed that the way that cognitive load is expressed in speech by subjects depends strongly on the task they had to perform. In Schuller et al. (2014), this could be confirmed by experimental results. Consequently, the basic approach here is that models are trained and evaluated for each sub-partition (CLSE-Time, CLSE-Dual, and CLSE-Span) individually.

In the development phase, i.e, the evaluation on the development set, the performance of the following hand-crafted feature sets was of interest: CL-Extended, CL-Base, and CL-Stress (Section 5.1). The unbalanced class distribution of CLSE-Span was adjusted by increasing the number of minority class instances in the training set using SMOTE (Section 6.1.1) in WEKA 3 (Witten et al., 2016); results obtained by other over-sampling techniques were generally lower in preliminary experiments. Some of the features contained in the above-mentioned feature sets could be irrelevant or redundant. Therefore, apart from using hand-crafted feature sets, feature selection was performed to obtain the most relevant features in an automatic manner. It was decided to use filters instead of wrappers in order to avoid model overfitting; the CL-Extended feature set in connection with the training data served as the basis for exploring the feature space. The following three filter methods were considered: feature ranking by information gain (IG), feature ranking by correlation (CC_P), and the correlation-based feature selection (CFS) algorithm (Section 6.1.3.2). In the case of IG and CC_P, an incremental evaluation of the top N features was performed on the development set to determine the global maximum in model performance; only those features were considered that have a relevance score > 0. However, large-scale feature sets may include many redundant features that do not contribute to a better performance. The CFS method addresses this issue by searching for a feature subset in which each feature is highly correlated with the target class while the correlation of the features among each other is low.

In the experiments, three feature normalisation methods were investigated, namely speaker normalisation, partition normalisation, and training normalisation

(Section 6.1.2.2). Since subject information is not included in the test set, the utilisation of speaker normalisation implies that speakers have to be identified in an automatic manner before feature values can be adjusted in terms of the speaker-context. A common approach to discriminate between speakers from a given input audio stream is to apply 'speaker diarisation'. The goal of a speaker diarisation system is to provide an answer to the question—'Who spoke when?'. In essence, such a system performs three main tasks: (1) discriminate between speech and non-speech regions; (2) detect speaker changes to segment the audio data; (3) fuse segmented data into speaker-homogeneous clusters (Shao et al., 2010, p. 30). In this thesis, the open-source toolkit LIUM SpkDiarization (Meignier and Merlin, 2010) was chosen for speaker identification due to its low error rates compared to other diarisation toolkits (cf. Kiktova and Juhar, 2015). After summarising all test set instances per task into a single audio stream and applying speaker diarisation, 14 clusters for CLSE-Time, 13 clusters for CLSE-Dual, and 8 clusters for CLSE-Span were obtained. Subsequently, feature values were normalised on a per cluster basis, which corresponds to (pseudo) speaker normalisation. As pointed out in Section 3.1.2, the actual number of subjects in the test set is 8; each of them participated in all tasks.

Further, two static classification methods were contrasted: support vector machines (SVMs) and multilayer perceptrons (MLPs). SVMs with polynomial kernel functions were trained using the Sequential Minimal Optimisation (SMO) algorithm (Keerthi et al., 2001), which is implemented in the WEKA 3 data mining toolkit (Witten et al., 2016). The SVM complexity parameter C was optimised on the development set. This was achieved by evaluating $C = 10^n$, with $n = -5, -4, \ldots, 0$. Higher values were omitted to avoid overfitting. Regarding the second classification method, MLPs were employed with one hidden layer for which sigmoid activation functions were used. For the output layer, the softmax activation function was chosen. The Java-based toolkit DeepLearning4J (Deeplearning4j Team, 2016) was used for building the networks. Constant parameter setting was determined empirically: 50 epochs were used for model training at a learning rate of 0.01 and the momentum (Nesterov) was set to 0.9. For optimisation, the number of hidden nodes HN was investigated on the development set by $HN = 2^n$, with $n = 3, 4, \ldots, 9$. In the case of CLSE-Span, the number of instances is much higher compared to the other tasks. In turn, this is more demanding in terms of the computation time for model training. In order to compensate for this fact, mini-batches were used with a size of four instances—which was computationally more efficient.

In summary, before a selection of systems is evaluated on the independent test set, different configurations were investigated in the development phase. Six feature sets (three predefined sets and three sets obtained by feature selection), three feature normalisation methods, six SVM-hyperparameter values and seven

MLP-hyperparameter values were evaluated. All three tasks were considered, which yields a total of 702 results.

6.6.2 Results

In this section, different system configurations are evaluated according to the rules of the Cognitive Load Sub-Challenge. Three levels of cognitive load (L_1: low, L_2: medium, and L_3: high) had to be classified automatically. Each cognitive load task (CLSE-Time, CLSE-Dual, and CLSE-Span) was considered to be independent. Initially, the training set was used for feature selection and model training; parameter optimisation was performed on the development set. In the following, experimental results are reported for optimal classifier parameters. For information on hyperparameter settings, the reader is referred to Appendix B.4.

First, feature selection methods are evaluated in connection with feature normalisation strategies (SN: speaker normalisation, PN: partition normalisation, and TN: training normalisation) and classifiers (SVM: support vector machine and MLP: multilayer perceptron). Figure 6.7 illustrates the UAR performance according to the number of ranked features by information gain (IG). For each system configuration, classifier parameters were constant across the number of features—results are shown for the best global maxima. Since the basic criterion was IG > 0, the number of features is restricted: 502 (SN) and 205 (PN/TN) for CLSE-Time; 205 (SN) and 33 (PN/TN) for CLSE-Dual; 277 (SN) and 275 (PN/TN) for CLSE-Span. For reasons of clarity, only the top 250 features are shown. Incremental evaluation was performed with a step size of ten features. Regarding CLSE-Time, it can be seen that SVM performs well when using SN and the top 130 features (85.71% UAR, 11.11% above the baseline). The best UAR obtained by PN-SVM (top 70 features) corresponds exactly to the baseline, while all results obtained by TN-SVM remain below it. By using MLP on CLSE-Time, results above the baseline can be reached when SN or TN is applied. It is remarkable that the baseline can be outperformed with only 20 features (TN-MLP, 1.59% over the baseline). The best result for MLP on CLSE-Time (84.13% UAR) is achieved when the top 230 speaker-normalised features are used. In the case of CLSE-Dual, SVM with the top 40 features and MLP with the top 30 features, both with SN, yield highest UAR (73.02%), which corresponds to an improvement of 9.52% compared to the baseline. Moreover, PN-MLP outperforms the baseline as well (top 30 features); the absolute difference is 1.58%. Concerning IG ranking on CLSE-Span, none of the methods is capable of improving the baseline system. Here, the highest UAR (59.37%) is obtained by SN-MLP and the top 130 features; 1.83% below the baseline.

The performance according to the number of ranked features by correlation (CC_P) is shown in Figure 6.8. In this case, a relevance score > 0 was obtained by

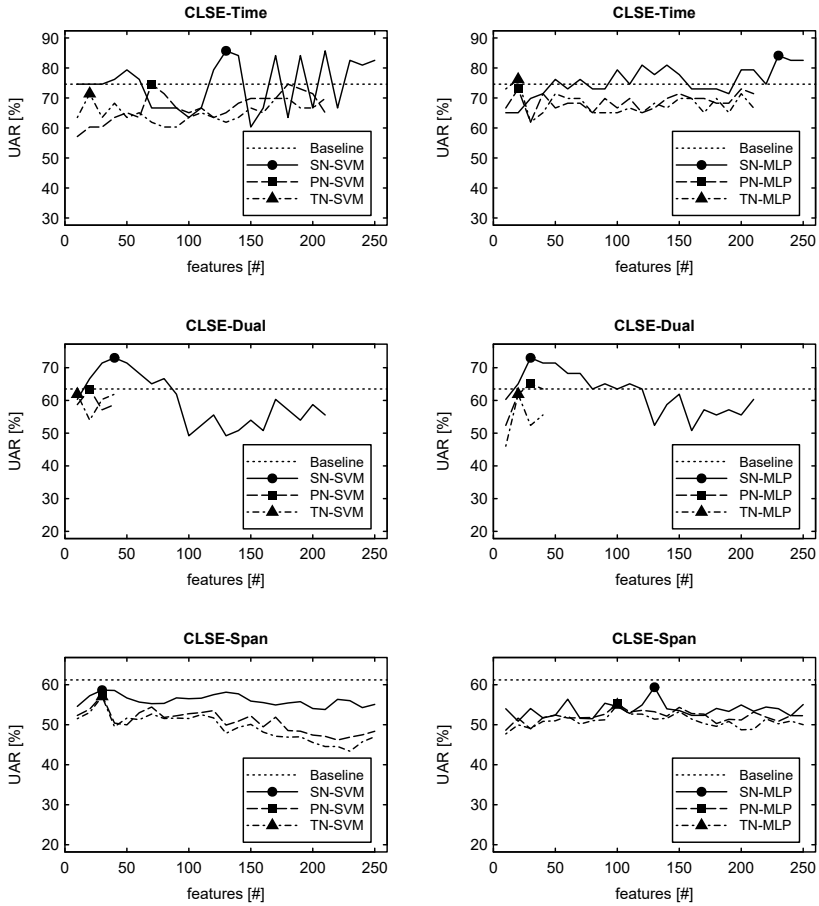


Figure 6.7 Classification results on the development set for CLSE-Time, CLSE-Dual, and CLSE-Span according to the number of ranked features by information gain obtained from the corresponding training set. Global maxima are highlighted. Comparison of feature normalisation methods (SN: speaker normalisation, PN: partition normalisation, and TN: training normalisation) and classifiers (SVM: support vector machine and MLP: multilayer perceptron)

almost all features. It was decided to restrict the number of top-ranked features to a maximum of 1,000. The step size for incremental evaluation was set to 50 features. As in the case of IG ranking, results are shown for classifier parameters that were constant across the number of features and superior in terms of the

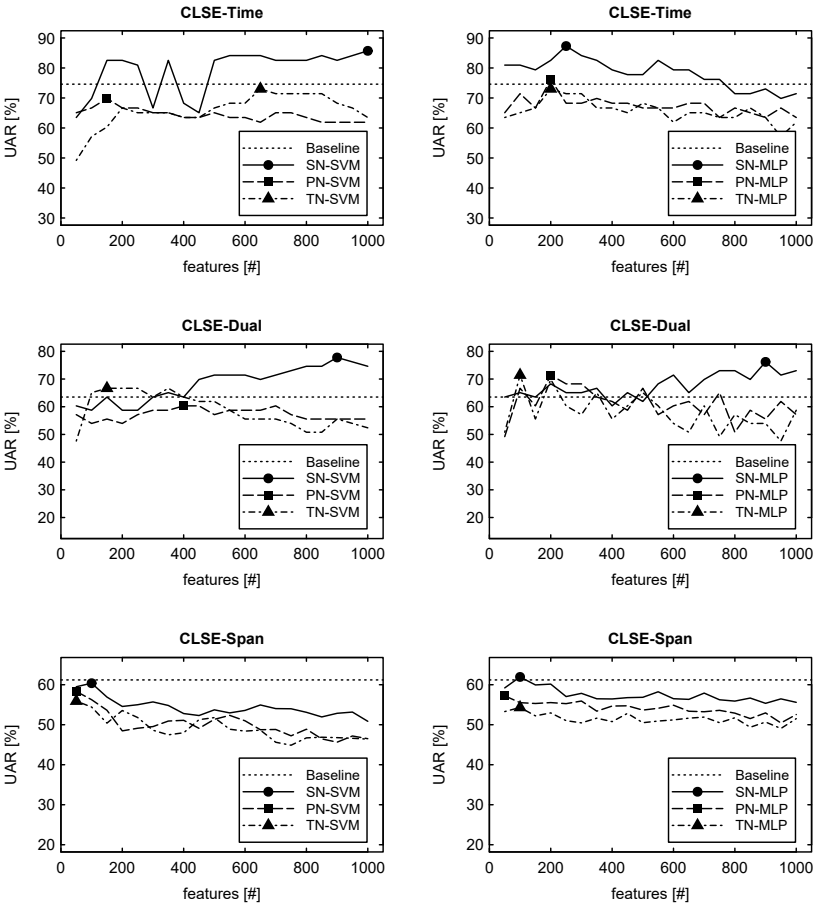


Figure 6.8 Classification results on the development set for CLSE-Time, CLSE-Dual, and CLSE-Span according to the number of ranked features by correlation obtained from the corresponding training set. Global maxima are highlighted. Comparison of feature normalisation methods (SN: speaker normalisation, PN: partition normalisation, and TN: training normalisation) and classifiers (SVM: support vector machine and MLP: multilayer perceptron)

global maximum of UAR. Regarding CLSE-Time, the SN-SVM classifier clearly outperforms the baseline system (best is at 1,000 features—85.71% UAR). The performance on CLSE-Time is further improved by SN-MLP with the top 250 features (87.30% UAR), which corresponds to an absolute difference of 12.7%

compared to the baseline system. MLP in combination with PN yields lower results, though the maximum (200 features) is still above the baseline. Looking at the results for CLSE-Dual, all global maxima—except that of PN-SVM—outperform the baseline system. The best SVM system performs slightly better than the best MLP system, the difference is 1.59%. For CLSE-Span, only SN-MLP seems to be promising (61.95% UAR at 100 features), although there is only little improvement over the baseline (0.75%).

Apart from the incremental evaluation for feature subset selection, the correlation-based feature selection (CFS) algorithm was applied to the training material in order to obtain feature subsets that are characterised by high feature relevance and low feature redundancy. Compared to the full feature set (cf. Section 5.1.1), feature subsets selected by CFS contain a relatively small number of features: 64 (SN) and 45 (PN/TN) for CLSE-Time; 54 (SN) and 20 (PN/TN) for CLSE-Dual; 31 (SN) and 26 (PN/TN) for CLSE-Span.

Table 6.15 summarises the results on the development set (UAR in % for each task individually) obtained by different feature sets, feature normalisation methods, and classifiers. Configurations that outperform the baseline system (CLSE-Time: 74.6%; CLSE-Dual: 63.5%; CLSE-Span: 61.2%) are highlighted in bold. Interestingly, feature selection methods are superior to the hand-crafted feature sets in 15 out of 18 configurations (task/normalisation/classifier). Only the SN-MLP classifier outperforms the development-baseline across all tasks.

According to the rules of the challenge, all three tasks have to be considered in terms of the development and test set. For this purpose, combinations of feature normalisation methods (SN/PN/TN) and classifiers (SVM/MLP) across tasks were created resulting in six different system configurations. Each system consists of three models, one model for each task. Each model is characterised by the best performing features determined in the development phase (cf. Table 6.15). For the evaluation on the test set, models were re-trained for each task individually using both the training and development set (hereinafter referred to as 'pooled training data'), since it was assumed that robustness is enhanced by involving more data for modelling. If one of the models was based on feature selection then the features obtained in the development phase were adopted, i.e., feature selection was not applied again. The class distribution of the pooled training data was balanced for CLSE-Span by SMOTE (Section 6.1.1) using WEKA 3 (Witten et al., 2016). The feature normalisation methods PN and TN were applied once again. Per speaker normalisation is only feasible for the pooled training data since subject IDs are not included in the test set. However, this fundamental problem can be eased by using speaker diarisation (cf. Section 6.6.1). In this way, speaker-clusters were obtained from the test set and then normalised individually—denoted hereinafter by 'speaker-cluster normalisation' (SCN). For modelling with the pooled training data, the optimal parameter settings

Table 6.15 Development results per feature set, feature normalisation method, and classifier for each task of COMPARE 2014 (Cognitive Load Sub-Challenge)

		UAR [%]					
		SVM			MLP		
Task	Features	SN	PN	TN	SN	PN	TN
CLSE-Time	CL-Extended	**84.13**	63.49	65.08	**76.19**	71.43	68.25
CLSE-Time	CL-Base	**87.30**	**77.78**	68.25	**85.71**	73.02	68.25
CLSE-Time	CL-Stress	**84.13**	68.25	68.25	**85.71**	71.43	66.67
CLSE-Time	IG*N*	**85.71**	74.60	71.43	**84.13**	73.02	**76.19**
CLSE-Time	CC$_P$*N*	**85.71**	69.84	73.02	**87.30**	**76.19**	73.02
CLSE-Time	CFS	**80.95**	68.25	66.67	**80.95**	60.32	60.32
CLSE-Dual	CL-Extended	**69.84**	**65.08**	**65.08**	**74.60**	**68.25**	**66.67**
CLSE-Dual	CL-Base	**69.84**	53.97	53.97	**73.02**	57.14	63.49
CLSE-Dual	CL-Stress	**65.08**	55.56	58.73	**69.84**	60.32	61.90
CLSE-Dual	IG*N*	**73.02**	63.49	61.90	**73.02**	**65.08**	61.90
CLSE-Dual	CC$_P$*N*	**77.78**	60.32	**66.67**	**76.19**	**71.43**	**71.43**
CLSE-Dual	CFS	61.90	63.49	**65.08**	58.73	**65.08**	**66.67**
CLSE-Span	CL-Extended	54.85	49.68	48.01	55.78	47.83	49.39
CLSE-Span	CL-Base	55.39	54.38	49.37	55.08	55.06	50.58
CLSE-Span	CL-Stress	55.70	53.60	46.92	57.99	54.48	50.34
CLSE-Span	IG*N*	58.65	57.33	57.06	59.37	55.23	54.94
CLSE-Span	CC$_P$*N*	60.38	58.32	55.91	**61.95**	57.23	54.32
CLSE-Span	CFS	60.98	58.54	56.44	58.98	54.01	52.77

Results above the development-baseline are highlighted in bold. Abbreviations: MLP (multilayer perceptron), PN (partition normalisation), SN (speaker normalisation), SVM (support vector machine), TN (training normalisation), UAR (unweighted average recall)

determined in the development phase were used. The predictions from the task-dependent evaluations (development and test set) were concatenated and scored for UAR normally.

The final results are briefly summarised in Table 6.16. As mentioned above, SCN is only applied to the test set. Looking at the results, it can be seen that systems with speaker normalisation (SN) outperform the baseline on the development set, while the remaining normalisation methods stay behind. The SN/SCN-SVM system yields a UAR of 65.5% (a plus of 2.3%), but it performs worse than the baseline system on the test set. The highest UAR on the development set is achieved by SN/SCN-MLP (69.7% UAR), 6.5% above the baseline system. Furthermore, SN/SCN-MLP outperforms the baseline on the test set with 62.2% UAR, which corresponds to an absolute difference of 0.6%. It can be

Table 6.16 Summary of the final results for COMPARE 2014 (Cognitive Load Sub-Challenge)

			UAR [%]	
Features	Norm	Classifier	Devel	Test
CL-Base (T), CC$_P$*N* (D), CFS (S)	SN/SCN	SVM	**65.5**	58.8
CC$_P$*N* (T), CC$_P$*N* (D), CC$_P$*N* (S)	SN/SCN	MLP	**69.7**	**62.2**
CL-Base (T), CL-Extended (D), CFS (S)	PN	SVM	61.1	59.4
CC$_P$*N* (T), CC$_P$*N* (D), CC$_P$*N* (S)	PN	MLP	60.5	56.8
CC$_P$*N* (T), CC$_P$*N* (D), IG*N* (S)	TN	SVM	59.4	54.1
IG*N* (T), CC$_P$*N* (D), IG*N* (S)	TN	MLP	58.5	55.3
Baseline system (Schuller et al., 2014)			63.2	61.6

Features are shown for each task; tasks are given in brackets. Results above the baseline are highlighted in bold. Abbreviations: D (CLSE-Dual), Devel (development set), MLP (multilayer perceptron), Norm (feature normalisation method), PN (partition normalisation), S (CLSE-Span), SCN (speaker-cluster normalisation), SN (speaker normalisation), SVM (support vector machine), T (CLSE-Time), Test (test set), TN (training normalisation), UAR (unweighted average recall)

Table 6.17 Confusion matrix for the test set of COMPARE 2014 (Cognitive Load Sub-Challenge) showing classification in percentage for low (L_1), medium (L_2), and high (L_3) cognitive load by the best system configuration

		predicted			
		L_1	L_2	L_3	$\sum$
actual	L_1	76.4%	13.9%	9.7%	216
	L_2	13.9%	62.5%	23.6%	216
	L_3	16.7%	35.6%	47.8%	312

concluded that, theoretically, the sixth place among all participants is reached (cf. Table 6.2 in Section 6.1.6). Eight participants (including the baseline) submitted system results at the time of the challenge; only seven of them reported test set results in the corresponding papers.

It was shown that filter-based feature selection methods are superior to predefined feature sets and adjusting the feature values by speaker normalisation (and speaker-cluster normalisation) helps to improve the classification performance. Some details on the test set results obtained by the best proposed system (SN/SCN-MLP) are given in Table 6.17. The cognitive load levels L_2 and L_3 are

confused to some degree. The best performance is obtained for L_1, whereas L_3 seems to be more challenging.

6.7 Regression Approaches

In the previous sections, various features and algorithms have been extensively evaluated for the automatic recognition of cognitive load, whereby the load has been considered a classification problem in terms of three different levels (low, medium, and high).

In this section, regression-based cognitive load recognition from speech is proposed for the first time—here, the system outputs are continuous numerical values instead of discrete categories. Different feature sets, feature normalisation methods, and regression algorithms are evaluated on the CoLoSS corpus (Section 3.2). In this connection, modelling is related to the efficiency score of the secondary task. As pointed out in Section 3.2.4, the higher the cognitive load imposed by the primary task, the lower the efficiency in the secondary task. Two questions are investigated: (1) 'Which feature set is useful for modelling?'; (2) 'What is the effect of modelling temporal dependencies regarding the chronological sequence of task trials?'.

In the following, the experimental setup is presented (Section 6.7.1) and evaluation results obtained are discussed (Section 6.7.2).

6.7.1 Experimental Setup

Regression experiments were conducted on the CoLoSS corpus (Section 3.2), whereby modelling of cognitive load refers to the secondary task efficiency (Eff_{ST}) scores. The Eff_{ST} values range between 0 and 0.86; for further information on the data, see Section 3.2.5. A leave-one-subject-out cross-validation (LOSOCV) was chosen as an evaluation strategy to guarantee speaker-independent conditions and to obtain a reliable estimate of the model performance. The experimental methodology used here follows that of Section 6.2. Of course, class balancing is not needed.

In the experiments, the following three proposed feature sets were contrasted: CL-Extended, CL-Base, and CL-Stress (Section 5.1). In addition, the effectiveness of automatically selected features was of interest. To this end, the correlation-based feature selection (CFS) method (Section 6.1.3.2) was applied to CL-Extended. Three different approaches for the normalisation of the feature values were evaluated. These are speaker normalisation, partition normalisation, and training normalisation (Section 6.1.2.2).

For modelling using the extracted statistical features, support vector regression (SVR), multilayer perceptron (MLP) networks, and Long Short-Term Memory (LSTM) recurrent neural networks (RNNs) were employed. SVR was applied with a polynomial kernel function. This regressor was optimised in the same way as it was done for the support vector machine classifier in the previous evaluations. That is, the complexity C was tuned for each configuration by investigating $C = 10^n$, with $n = -5, -4, \ldots, 0$. For SVR model training, the Sequential Minimal Optimisation (SMOreg) algorithm (Shevade et al., 2000) provided by the WEKA 3 data mining toolkit (Witten et al., 2016) was used. Next, one-hidden-layer MLP networks with the sigmoid activation function for hidden layer neurons were evaluated. The parameter set was determined empirically based on preliminary experiments—stochastic gradient descent optimisation with Nesterov momentum (0.9) was applied at a learning rate of 0.01 for 20 epochs with a mini-batch size of 8. The influence of temporal context on the prediction performance was investigated by using LSTM RNNs. LSTM networks with one hidden layer were evaluated; the tanh function was used for the LSTM memory blocks. For temporal context modelling, the sequence length was set to eight consecutive task trials, where the current trial (prediction) corresponds to the last step in the sequence. In this way, a kind of sliding window was applied along the trial axis for each subject. With regard to the first trials of an overall trial series, the sequences had to be truncated as long as they did not reach the predefined length of eight. Only a few learning parameters differ from those of the MLP networks; here, 50 epochs were used with a mini-batch size of 32. MLP and LSTM networks were optimised by investigating different values for the number of hidden nodes (or memory blocks): 2^n, with $n = 3, 4, \ldots, 8$. Since regression is performed, the output layer of both networks consists of a single node to which the identity function was applied. For neural network design and model training, the Java-based toolkit DeepLearning4J (Deeplearning4j Team, 2016) was used.

Two performance measures were of interest. The Spearman's correlation coefficient (CC_S) and the root mean square error (RMSE) were chosen, whereby CC_S was considered to be the primary measure; it reflects the relationship between the actual and predicted values.[2] The results from the cross-validation folds were averaged to obtain a single estimate for each of the two measures.

The experiments can be summarised as follows: LOSOCV results for regression-based cognitive load recognition were produced on the CoLoSS corpus. Four feature sets (three predefined sets and one varying set obtained by feature selection), three feature normalisation methods, and six hyperparameter values were investigated in connection with three different regressors. In total, this yields 216 results.

[2]The CC_S was preferred for measuring the relationship because the Eff_{ST} data of 51 subjects is non-normally distributed (Shapiro-Wilk test, $p < 0.05$).

6.7.2 Results

A series of experiments was conducted to evaluate the prediction performance of regression methods. The goal was to automatically estimate the secondary task efficiency (Eff_{ST}), which reflects the cognitive load in a learning task scenario (Section 3.2.2). Features, feature normalisation methods, and regression algorithms are compared. The results obtained by each system configuration are reported in terms of tuned regressor-hyperparameters (see Appendix B.5).

Table 6.18 summarises the performance for all configurations; the Spearman's correlation coefficient (CC_S) is considered to be the primary evaluation measure. Regarding hand-crafted feature sets, CL-Base performs slightly better in general (in five out of nine cases). However, feature subsets obtained by the correlation-based feature selection (CFS) method include a much smaller number of features (SN—speaker normalisation: $\mu = 52.1$, $\sigma = 1.9$; PN—partition normalisation/TN—training normalisation: $\mu = 57.2$, $\sigma = 4.4$) and outperform all three predefined feature sets. This suggests that the regression methods are highly sensitive to both the number and the relevance of features. The differences between feature normalisation approaches are not very large. The best RMSE results are generally dominated by the TN method. From the table it can be

Table 6.18 Results for the automatic prediction of the secondary task efficiency

		SN		PN		TN	
Feature set	Regressor	CC_S	RMSE	CC_S	RMSE	CC_S	RMSE
CL-Extended	SVR	0.470	0.089	0.498	0.092	0.501	0.070
CL-Base	SVR	0.475	0.088	0.501	0.094	0.495	0.072
CL-Stress	SVR	0.475	0.088	0.499	0.095	0.496	0.071
CFS	SVR	0.530	0.087	0.533	0.095	**0.538**	**0.068**
CL-Extended	MLP	0.297	0.115	0.382	0.111	0.369	0.085
CL-Base	MLP	0.419	0.096	0.446	0.102	0.421	0.074
CL-Stress	MLP	0.367	0.103	0.426	0.105	0.416	0.078
CFS	MLP	0.504	0.090	0.509	0.094	0.490	0.070
CL-Extended	LSTM	0.079	0.152	0.045	0.197	0.081	0.129
CL-Base	LSTM	0.047	0.156	0.024	0.210	0.087	0.143
CL-Stress	LSTM	0.078	0.164	0.070	0.210	0.075	0.128
CFS	LSTM	0.078	0.137	0.064	0.157	0.105	0.107

Best result is highlighted in bold. Abbreviations: CC_S (Spearman's correlation coefficient), LSTM (Long Short-Term Memory), MLP (multilayer perceptron), PN (partition normalisation), RMSE (root mean square error), SN (speaker normalisation), SVR (support vector regression), TN (training normalisation)

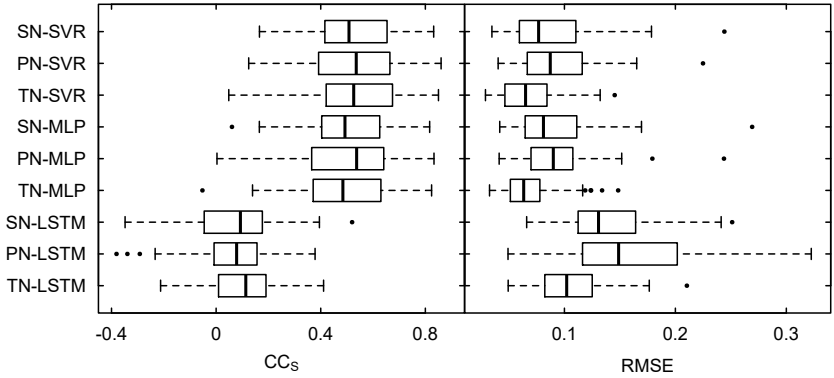


Figure 6.9 Boxplots of regression results per model (learned with CFS features)

seen that Long Short-Term Memory (LSTM) networks perform much worse than support vector regression (SVR) and multilayer perceptrons (MLPs). The overall best performance is obtained by SVR with CFS features and the TN method ($CC_S = 0.538$, $RMSE = 0.068$). This system performs significantly better than the best MLP (PN, CFS) and LSTM system (TN, CFS) in terms of CC_S as well as RMSE.[3]

Details on leave-one-subject-out cross-validation (LOSOCV) results obtained with CFS features are given in Figure 6.9—the distributions per model are shown as boxplots. Regarding CC_S results, SVR and MLP show similar characteristics, while LSTM models stay behind. The highest median of CC_S is observed for PN-MLP (0.537) and the maximum of single CC_S results across all models is obtained by PN-SVR (0.860). Looking at the RMSE, results differ considerably and outliers are present for almost all models. MLP with TN shows the lowest median of RMSE (0.063), while SVR with TN yields the lowest RMSE (0.029). LSTM networks show less stable RMSE results, in particular when features have been normalised by the PN method.

To provide an impression of how the best system works, predicted and actual values are visualised for four subjects in Figure 6.10. The grey line indicates the system outputs, whereby each output refers to a single task trial. In this respect, it is important to remember that the higher the cognitive load, the lower the Eff_{ST} (cf. Section 3.2.4). Judging from the plots, there is a clear need for improvement regarding the range of predicted values. While the regressor performs well for

[3] Statistical significance was tested using a one-sided paired t-test for the results produced on the folds of LOSOCV, with a significance level of $\alpha = 0.01$.

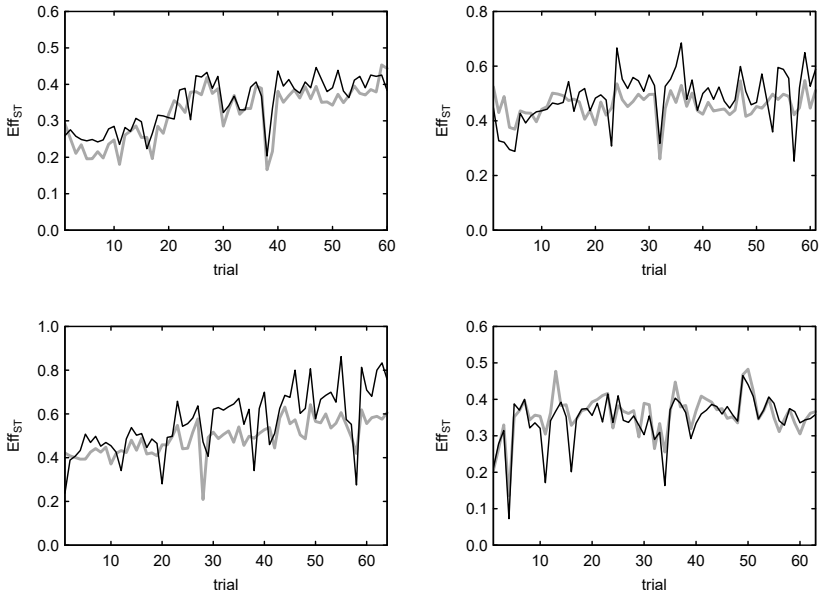


Figure 6.10 Automatic prediction of secondary task efficiency (Eff_{ST}) per trial using the best system configuration (see Table 6.18). The black line represents the actual values and the grey line represents the predicted values. Examples are shown for different subject IDs from the CoLoSS corpus: '17' (upper left), '22' (upper right), '43' (lower left), and '65' (lower right)

Eff_{ST} up to approximately 0.5 (see upper left and lower right in the figure), it is not capable of predicting much higher values (see upper right and lower left in the figure). The reason for this can be attributed to the fact that the data of Eff_{ST} is non-normally distributed (cf. Section 3.2.5)—there is not enough training material available regarding higher values so that the regression method has no chance to learn sufficiently.

It can be concluded that the best regression system for automatic cognitive load assessment is based on SVR and CFS features together with the TN method for feature normalisation. Further improvements might be achieved by better handling the given characteristics of the data distribution in the training sets, for example, by performing data transformation (e.g., logarithm function) or using additional data from similar tasks. The results obtained by LSTM networks showed that the data of the CoLoSS corpus is not suitable for modelling with contextual information from the past when testing, as well as scoring, is carried

out on a per subject basis.[4] The reasons for the much lower performance might be that: (1) there is no specified relationship between a current trial and the past trials concerning the primary task (random selection of symbol combinations, cf. Section 3.2.1), which, in turn, may cause fluctuations of cognitive load and patterns in speech; (2) the time span between two successive trials (approximately 20 seconds), and thus between two utterances, might be too long to derive meaningful information from the past—in previous works, LSTM networks were applied for speech processing on the basis of frames or utterances without such delays (cf. Wöllmer et al., 2008, 2010; Eyben et al., 2010). In addition, allowing LSTM networks to have access to the past as well as the future information—realised by bidirectional LSTM networks—might improve the performance.

[4]Note that LOSOCV without averaging per subject results generally yields higher correlation values. For instance, in this way, a CC_S of 0.404 can be achieved by TN-LSTM with CFS features, which corresponds to an absolute difference of 0.299 compared to the strict evaluation conducted in this thesis (cf. Table 6.18).

Chapter 7

Conclusion

This chapter summarises the research work presented in this thesis. First, the contents of this thesis are highlighted and it is discussed how the main aims have been achieved (Section 7.1). Finally, open issues and future directions are addressed (Section 7.2).

7.1 Summary

This thesis focuses on the effectiveness of paralinguistic speech features for the automatic assessment of cognitive load. The research presented throughout this work was motivated by open issues from past research efforts—important aspects, such as use-case scenarios, features, and modelling approaches, had not yet been touched sufficiently. This led to the definition of the following three aims for this thesis: (1) Development of a new speech-based cognitive load database, which reflects use-cases in which individuals are required to achieve learning goals; (2) Evaluation of traditional speech features and those from the field of stress detection for cognitive load recognition; (3) Development and evaluation of regression-based cognitive load analysis from speech.

For the investigations conducted, the speech material from two databases was used (Chapter 3). The first one, the CLSE database, was provided for this thesis and contains speech recordings of participants who performed three different tasks, each composed of three levels of cognitive load. Two variants of the Stroop test are included: Stroop test with time pressure (CLSE-Time) and Stroop test with dual-task (CLSE-Dual). The third task contained in CLSE is a reading span task (CLSE-Span), which is related to the working memory capacity by the maximum number of items that can be stored. The second database, called CoLoSS, was developed within the scope of this thesis. This database contains recordings of participants who performed a learning task while cognitive load was induced by the corresponding learning processes—thereby, aim (1) was fully achieved. Fundamentally speaking, CoLoSS contrasts with existing cognitive

load databases, since learning tasks have not yet been employed and it provides continuous numerical labels in addition to the discrete labels used until now.

A number of indicative speech features were extracted and discussed (Chapter 4). This particularly concerns prosodic features. Given the evidence for the importance of temporal aspects in speech related to the human working memory, it is essential to consider the corresponding prosodic parameters. To this end, a software framework for the extraction of duration-based features and a threshold-based activity detector for the extraction of speech-events were implemented in this thesis. Complementary information to the prosody was provided by extracted frequency content of the speech signal. From the literature it is observed that cognitive load can result in increased muscle tension of the human vocal folds. Hence, speech quality features were also taken into account. Although tasks may be associated with factors of stress, for example, due to time pressure or secondary task demands, speech features that are assumed to be suitable for stress detection have not yet been considered for cognitive load recognition. This gap has been closed in this thesis by investigating three promising Teager energy operator (TEO) based features. As a further contribution, feature sets were designed for static modelling of automatic cognitive load (CL) recognition (Section 5.1). The CL-Extended feature set, which contains a total of 3,201 statistical parameters, represents the amount of all speech features extracted for this thesis. Two feature subsets were derived, namely CL-Base and CL-Stress. While the former is motivated by studies from the scientific literature in terms of the effects of cognitive load on speech, the latter in which TEO features are included is intended for tasks that are associated with stress factors. For achieving aim (2), the effectiveness of the extracted speech features was successfully evaluated by feature relevance analyses and recognition experiments, as described in the following.

Feature relevance analyses were conducted to verify the suitability of speech features for use in different cognitive load tasks. Correlation results, presented in Section 5.2, confirmed the assumptions regarding the importance of the TEO for tasks that may be associated with stress, such as variants of the Stroop test (CLSE-Time and CLSE-Dual). On the other hand, prosodic features describing temporal aspects are generally relevant, in particular for the learning task employed in the CoLoSS corpus. An approach to the analysis of the generalisation across tasks revealed that the most effective features refer to the tempo in speech. Next, different aspects of speech were evaluated by the information gain (Section 5.3). In addition to the above-mentioned results, it was found that the intensity of the speech signal is a promising indicator in terms of the Stroop test variants. Moreover, for all datasets used in this thesis, the top 100 features ranked by the information gain showed again that prosodic features are particularly relevant and—interestingly—all tasks are dominated by TEO features.

A series of recognition experiments was carried out to investigate the effectiveness of speech features for automatic cognitive load assessment (Chapter 6). Speaker-independent systems were designed and extensively evaluated for three-class classification (low, medium, and high) and prediction by regression. Considering all configurations with respect to feature sets, feature normalisation methods, learning algorithms, and use-case scenarios, more than 3,000 results were produced.

Classification systems based on support vector machines (SVMs) were systematically evaluated for each cognitive load task individually (Section 6.3). Although TEO parameters showed excellent results in feature type comparison on the CLSE-Time dataset, the CL-Base features in which TEO parameters are not included are superior. For the data of CLSE-Dual and CLSE-Span, feature subsets obtained by feature selection algorithms achieved the best results. Regarding the learning task employed in the CoLoSS corpus, the most effective features refer to the prosody, in particular to the proposed duration-based features. It turned out that there are large differences in performance between tasks—while systems perform well for the variants of the Stroop test, both the reading span task and the learning task seem to be more challenging. For instance, the absolute difference in performance between CLSE-Time and CLSE-Span is 24.11%. Indeed, it is questionable to suggest a single outstanding system configuration due to the relatively small number of speech examples that were recorded under laboratory conditions. Hence, the most effective features were estimated by considering the top three systems per task. The outcome was that prosodic features and the CL-Base feature set perform best in general and that the latter should be preferred since it covers more aspects in speech. Moreover, per speaker standardisation is generally the best feature normalisation method.

Next, the transferability of SVM-based cognitive load classification systems was investigated by cross-corpus experiments (Section 6.4), i.e., task-specific models were evaluated on the speech data of a different cognitive load task. Here, the effectiveness of the three proposed hand-crafted feature sets was of interest. When testing is carried out across many different tasks at a constant SVM complexity, the CL-Base feature set exhibits the best generalisation properties. Nevertheless, the results are only little above the chance level (33.33%) and obviously too low for practical use. SVM complexity optimisation for each target task individually leads—as expected—to better results, even a plus of about 10%. But in this case, a clear trend regarding effective feature sets cannot be observed. Another finding is that task-specific systems perform better in domains that are related to the system-design to some extent; this can be proven, for example, by employing variants of the Stroop test for model evaluation. Attention was also given to the fusion of data from various tasks for modelling, while the test set represents a separate task. In this respect again, the CL-Base feature

set is superior on average, although the performance is comparable to that of single-task training. Apart from that, the so-called mixed-corpus methodology was introduced (Section 6.5), where the core idea was to involve various task-contexts for modelling, while at the same time knowledge about the target task is considered. This was realised by combining different corpora for SVM model training, whereby a subset of one out of these corpora is excluded for testing. Hand-crafted feature sets were compared, and this time, the CL-Stress feature set performs better in general, although the differences between results are rather small. Due to the influence of data from the target task, systems perform better than in cross-corpus settings, but the results are still below those from evaluations where training and test data refer to the same task. Therefore, it is strongly recommended to perform per task modelling and testing, if possible; otherwise, for modelling, one should collect as much data as possible from tasks that have characteristics similar to those of the target task.

Another aspect was addressed, which refers to the inter-speaker variability in use-case scenarios where, however, information about speakers is not present. As a solution to this issue, cognitive load classification systems were developed that take the advantages of speaker identification techniques for the normalisation of features, i.e., feature values were scaled on a per speaker-cluster basis (Section 6.6). By applying speaker-cluster normalisation on feature subsets obtained by filters and employing multilayer perceptrons for task-dependent classification, baseline results of the INTERSPEECH 2014 Cognitive Load Sub-Challenge could be outperformed—a recognition performance of 62.2% was reached, which corresponds to a difference of 0.6%.

Finally—for achieving aim (3)—regression-based cognitive load recognition systems were designed and evaluated (Section 6.7). For this purpose, the data collection of the CoLoSS corpus was used. Support vector regression (SVR), multilayer perceptrons (MLPs), and Long Short-Term Memory (LSTM) recurrent neural networks were compared in conjunction with the three proposed feature sets and feature subsets obtained by the correlation-based feature selection (CFS) algorithm. The results showed that CFS outperforms all three predefined feature sets. Another result is that the CoLoSS corpus is not suitable for modelling with contextual information from the past task trials, which was proven by the use of unidirectional LSTM networks. The overall best regression-based cognitive load assessment system is based on SVR. Its outputs are moderately correlated with the actual values; results of the best MLP system are comparable.

7.2 Future Work

Different open issues arose during the research work conducted. First, the speech material used in this thesis was recorded under laboratory conditions. Speaker and environmental variability have been minimised by using feature normalisation methods. In realistic scenarios, however, a number of additional factors might affect the applicability of a cognitive load monitoring system. For instance, different channel types can be used, such as telephone or different microphones. This issue becomes even more serious in an ambient noise context. Hence, databases that contain speech from a large number of speakers recorded under real-world conditions are needed to investigate the effects of environmental differences on both the user's cognitive load as well as the recognition performance of a system. It would also be interesting to explore strategies for taking the individual characteristics of users into account, for example, by clustering speakers according to their expertise level. Moreover, attention should be paid to data selection to better exploit the amount of training material available for modelling—one way of achieving this is to use measures of the distance between class centres (Schuller et al., 2011b).

Another aspect, which should be addressed, is related to the design of the proposed feature sets (Section 5.1). Teager energy features on its own have shown excellent results and performed even better than any other feature type in terms of Stroop test tasks in which users had to cope with stressful conditions (cf. Section 6.3.2). Nevertheless, traditional features represented by the CL-Base feature set perform better for such tasks than those of the CL-Stress feature set in which Teager energy features are included. Hence, for the future, the CL-Stress feature set should be refined to better benefit from stress-related features for cognitive load recognition.

In all classification experiments conducted in this thesis, cognitive load was considered a three-class problem. Probably some use-cases require a more abstract view of cognitive load. For instance, in critical working environments, a warning system could be employed for the purpose of detecting the operators' overload. This can be achieved by treating cognitive load as a binary classification problem: For the data used in this thesis, all speech examples representing low and medium load can be grouped to one class and those representing the high load to another one. Alternatively, one can easily divide the continuous numerical labels of the CoLoSS corpus (Section 3.2) into two intervals by equal-width binning.

Apart from that, linguistic features have proven to be potential indicators of the user's cognitive load. While the combination of linguistic and acoustic information has been demonstrated for emotion recognition (e.g., Lee et al., 2002; Schuller et al., 2004), the advantages of combining both information is still an

open issue for future work in the field of cognitive load recognition. Tasks such as reading comprehension may serve as a basis where the answers given by subjects may provide useful information due to the occurrence of spontaneous speech.

More recently, alternative approaches to audio analysis have gained attention, such as bag-of-audio-words (Bhatia et al., 2017), spectrogram representations for convolutional neural network architectures (Weißkirchen et al., 2017), and methods that operate directly on the raw time representation of a signal (Trigeorgis et al., 2016). It would be interesting to evaluate such approaches for automatic cognitive load recognition. Modelling with temporal context based on task trial sequences using Long Short-Term Memory recurrent neural networks have proven not to be suitable in conjunction with the corpus developed in this thesis (cf. Section 6.7.2); this topic deserves more attention once databases are created that contain speech together with detailed information on how cognitive load evolves over short-time intervals. Moreover, cognitive load recognition systems designed so far produced per utterance results, although it can be assumed that cognitive load might vary quickly and application-oriented systems could benefit from continuous recognition. Continuous recognition can be realised, for example, by making predictions on smaller units (e.g., words) using dynamic algorithms such as Hidden Markov Models. In turn, the underlying labelled data is required but its creation is usually associated with high effort: One way is probably to vary the level of cognitive load through the task difficulty in short-time intervals. Another approach could be to continuously label the speech data through the subjective opinion of individual raters or by using reliable indicators such as physiological measures.

All in all, it is still a vision of the future but already on the horizon: Reliable interactive systems being capable of adaptive to the user's cognitive processes probably become part of our life as research and technology progress. The achievements of this thesis contribute to the development of system components that are responsible for the automatic assessment of cognitive load from speech—the effectiveness of speech features has been demonstrated in terms of new perspectives including cognitive load in the context of learning and the applicability of parameters that are highly indicative of stress, as well as the use of regression models, for cognitive load recognition. Two key aspects outside the scope of this thesis are of tremendous importance and certainly interrelated for cognitive load monitoring in real-life use-cases: real-time processing and multimodality. Substantial efforts were made in these directions, for example, the development of computational methods for real-time feature extraction (e.g., Eyben, 2015) and, on the other hand, the exploration of robust multimodal cognitive load measurement with physiological and behavioural modalities (e.g., Chen et al., 2016). Thereby, the basis for future research by the author of this thesis can be considered to be given.

Appendix A

Statistics

Statistics of various speech parameters were calculated on the CLSE database (Section 3.1) and the CoLoSS corpus (Section 3.2). The inter-speaker variability and task-dependent effects were minimised by performing z-score transformation on a per speaker basis for each task individually. Regarding the CLSE database, only the recordings of the training and development set were used, because information about speakers is not included in the test set.

Appendix A.1 presents means and 95% confidence intervals of speech parameters under different levels of cognitive load (L_1: low, L_2: medium, and L_3: high). In the case of CoLoSS, discretised labels were used (cf. Section 3.2.5). Statistical significance in terms of differences between load levels was determined through pairwise *t*-tests, with a Bonferroni-adjusted significance level of $\alpha = 0.05$. For parameters that contain multiple channels, the relationship to cognitive load was determined by correlation measurements per feature dimension (Appendix A.2). Here, the numerical labels of CoLoSS were used. Due to the ordinal nature of cognitive load labels of the CLSE database as well as non-normal distributed labels of the CoLoSS corpus, the Spearman's rank correlation coefficient was chosen. For the CoLoSS corpus, the sign of the coefficient has been changed due to the assumptions in terms of cognitive load (cf. Section 3.2.4). The significance of the correlation was tested (*t*-test) with the null hypothesis that the relationship between the speech parameter and the level of cognitive load is zero. Statistical results including significances are discussed in the corresponding sections of Chapter 4. The rule for interpreting the size of correlation is provided by Hinkle et al. (2003). Finally, Appendix A.3 presents the top 100 features ranked by the information gain along with the average of absolute, binary class correlations computed using Pearson's equation.

A.1 Means and Confidence Intervals

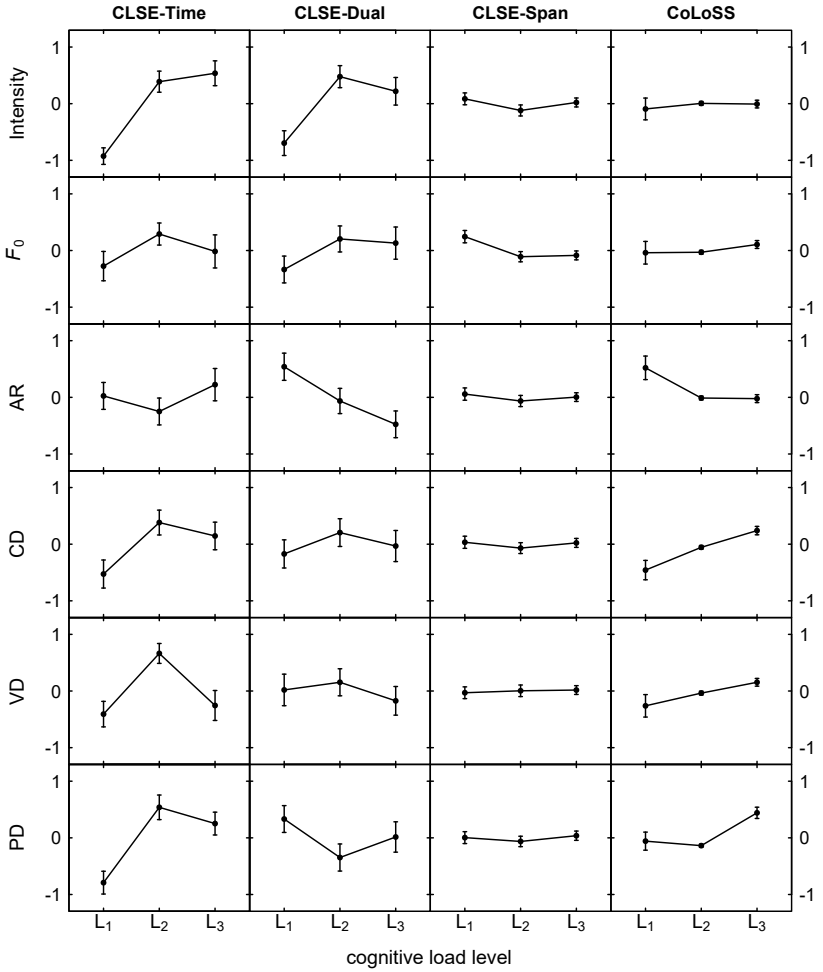


Figure A.1 Means and 95% confidence intervals of prosodic features under different levels of cognitive load per task. Abbreviations: AR (syllable-based articulation rate), CD (consonant duration), F_0 (fundamental frequency), PD (silent pause duration), VD (vowel duration)

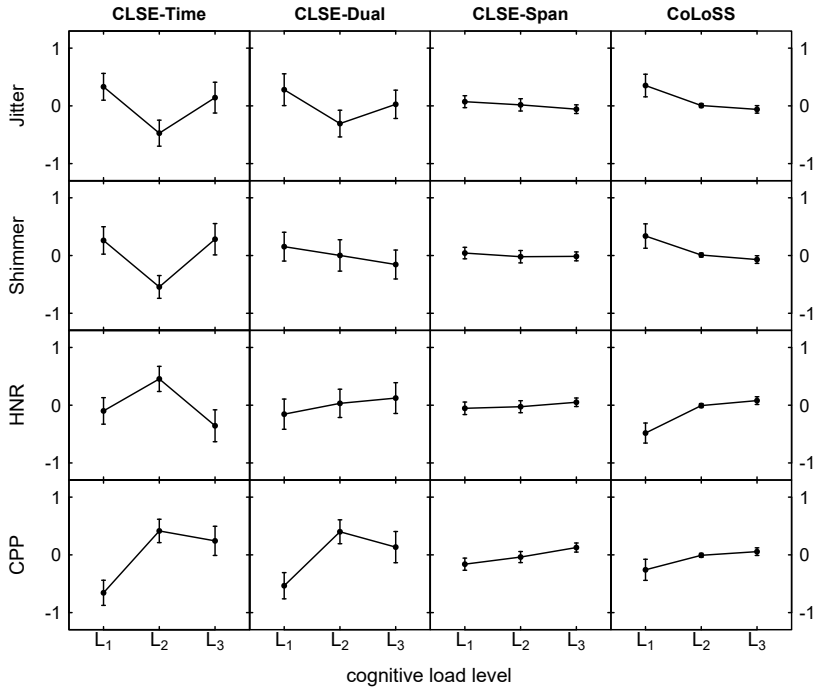


Figure A.2 Means and 95% confidence intervals of voice quality features under different levels of cognitive load per task. Abbreviations: CPP (cepstral peak prominence), HNR (harmonics-to-noise ratio)

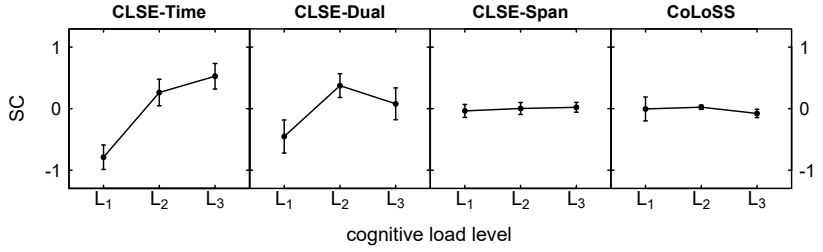


Figure A.3 Means and 95% confidence intervals of the spectral centroid (SC) under different levels of cognitive load per task

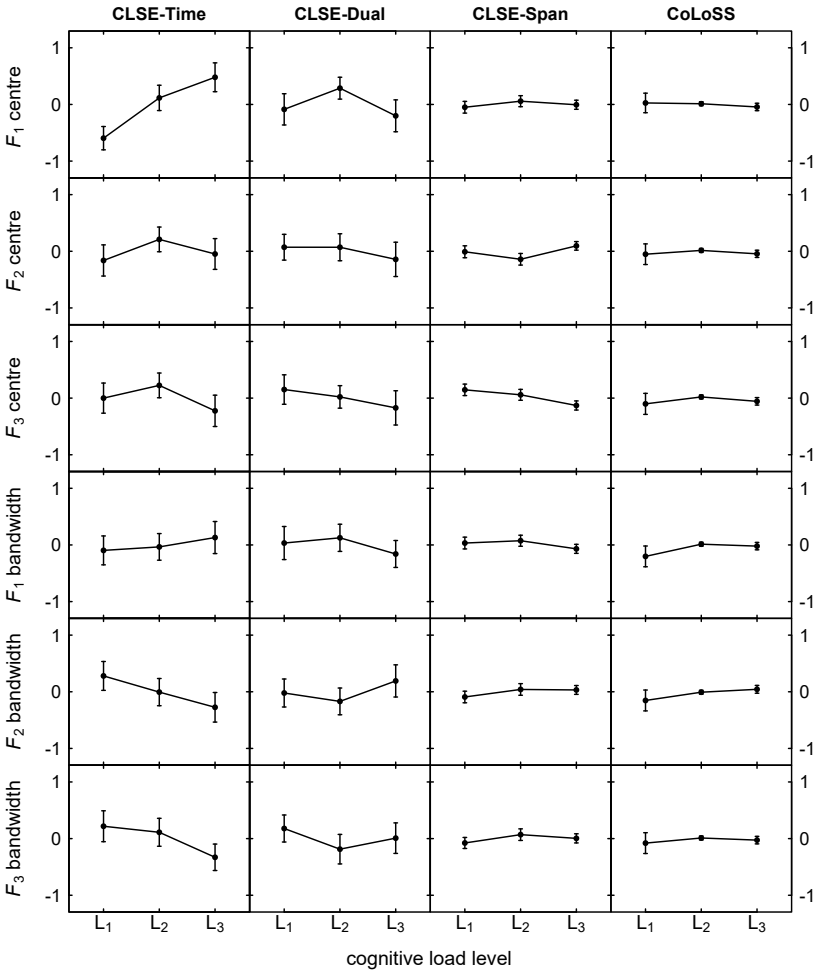
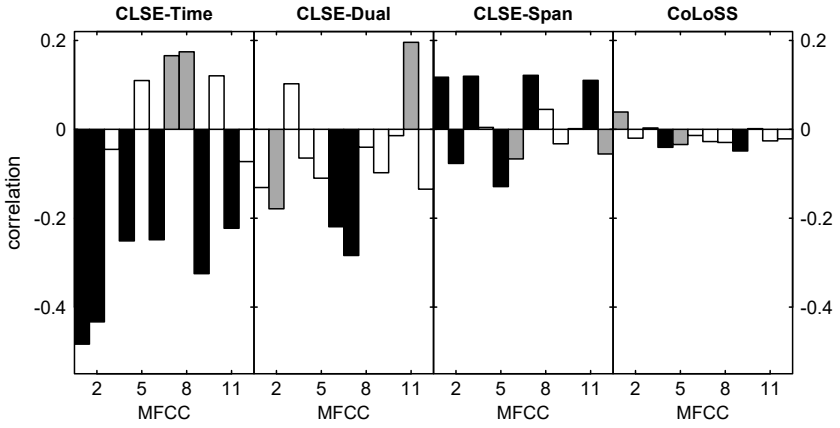
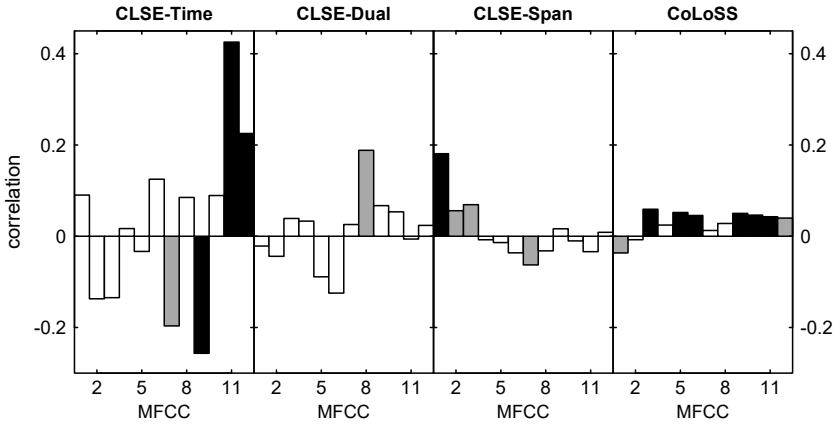


Figure A.4 Means and 95% confidence intervals of formants (centre frequency and bandwidth) under different levels of cognitive load per task

A.2 Channel Dependent Correlations

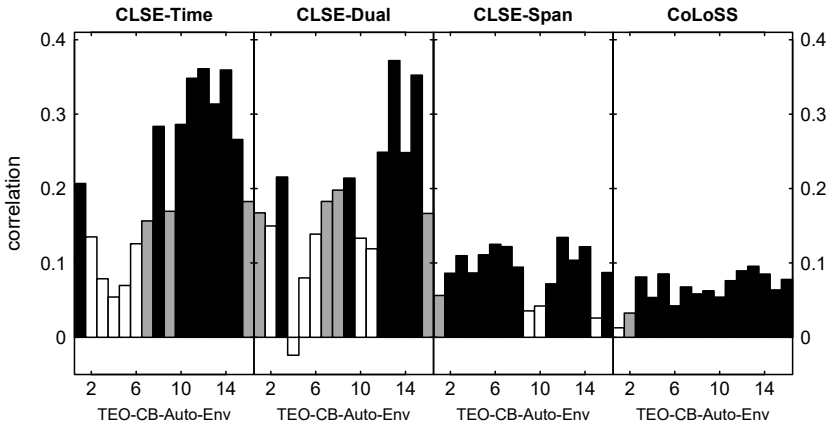


(a) mean of each MFCC per instance

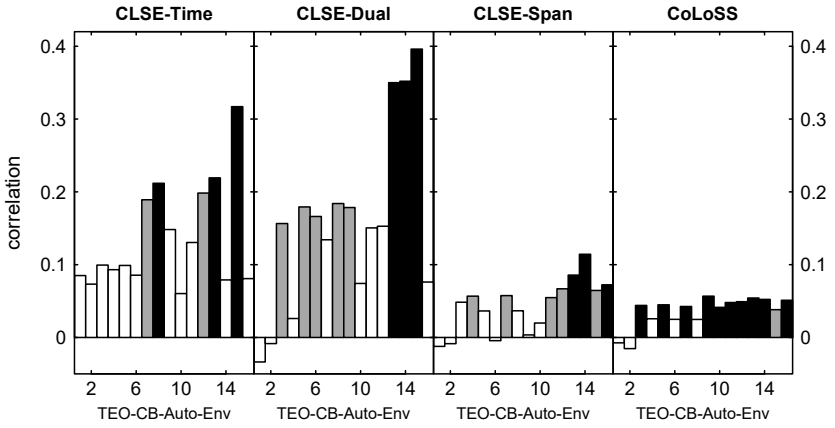


(b) standard deviation of each MFCC per instance

Figure A.5 Correlations between the level of cognitive load and 12 MFCCs per task; grey: significance ($p < 0.05$), black: strong significance ($p < 0.01$)

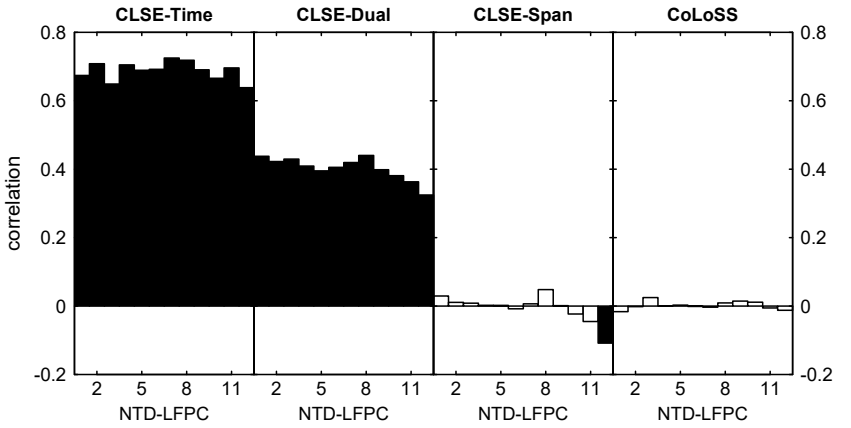


(a) mean of each TEO-CB-Auto-Env channel per instance

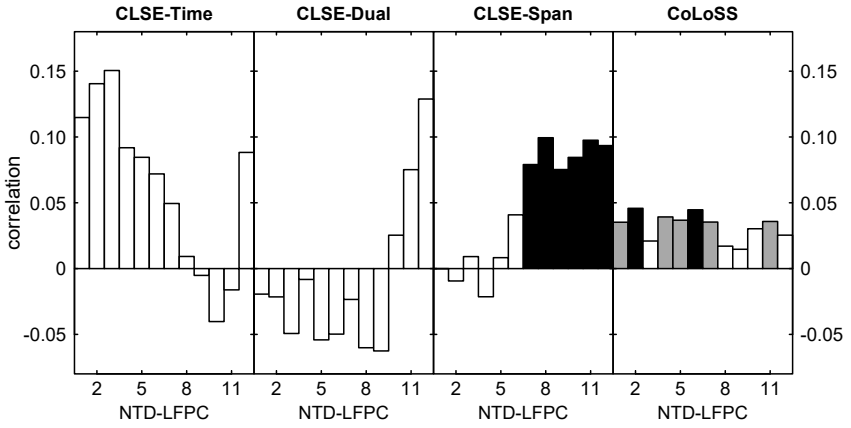


(b) standard deviation of each TEO-CB-Auto-Env channel per instance

Figure A.6 Correlations between the level of cognitive load and 16 TEO-CB-Auto-Env channels per task; grey: significance ($p < 0.05$), black: strong significance ($p < 0.01$)

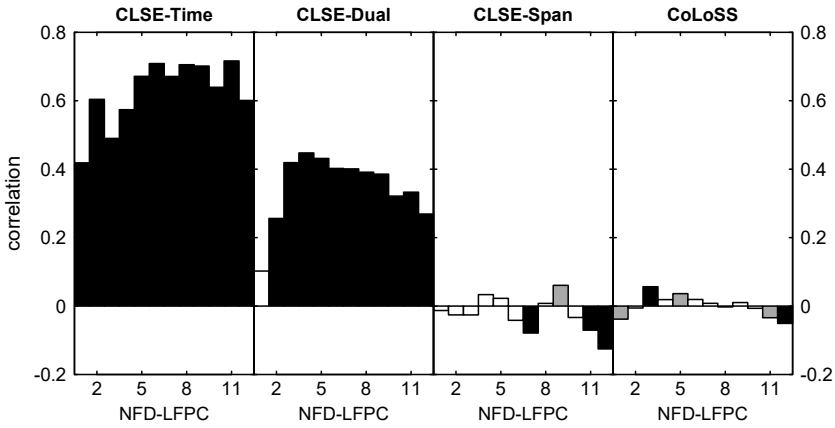


(a) mean of each NTD-LFPC per instance

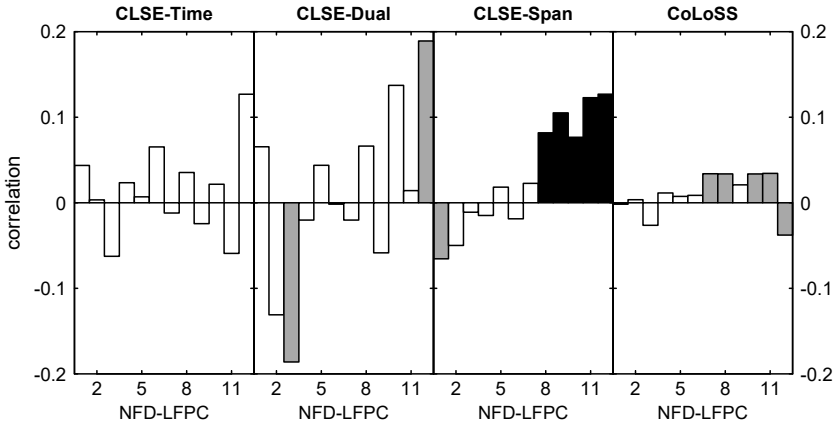


(b) standard deviation of each NTD-LFPC per instance

Figure A.7 Correlations between the level of cognitive load and 12 NTD-LFPC channels per task; grey: significance ($p < 0.05$), black: strong significance ($p < 0.01$)



(a) mean of each NFD-LFPC per instance



(b) standard deviation of each NFD-LFPC per instance

Figure A.8 Correlations between the level of cognitive load and 12 NFD-LFPC channels per task; grey: significance ($p < 0.05$), black: strong significance ($p < 0.01$)

A.3 Feature Ranking

Table A.1 Top 100 features ranked by information gain for CLSE-Time

Rank	IG	CC_P	Feature
1	0.720	0.522	NTD-LFPC 7 (Mean)
2	0.654	0.521	NTD-LFPC 2 (Mean)
3	0.645	0.499	NTD-LFPC 5 (75th percentile)
4	0.644	0.527	NTD-LFPC 8 (Mean)
5	0.644	0.505	NFD-LFPC 11 (25th percentile)
6	0.636	0.496	NTD-LFPC 1 (Mean)
7	0.632	0.499	NTD-LFPC 7 (75th percentile)
8	0.619	0.500	NTD-LFPC 11 (75th percentile)
9	0.612	0.506	NTD-LFPC 2 (50th percentile)
10	0.607	0.499	NTD-LFPC 4 (75th percentile)
11	0.605	0.509	NTD-LFPC 9 (75th percentile)
12	0.594	0.483	NTD-LFPC 1 (50th percentile)
13	0.593	0.487	NTD-LFPC 5 (50th percentile)
14	0.591	0.510	NTD-LFPC 6 (Mean)
15	0.591	0.520	NTD-LFPC 4 (Mean)
16	0.590	0.511	NTD-LFPC 5 (Mean)
17	0.587	0.487	NTD-LFPC 3 (Mean)
18	0.587	0.506	NTD-LFPC 7 (50th percentile)
19	0.586	0.511	NFD-LFPC 11 (Mean)
20	0.586	0.518	NTD-LFPC 11 (Mean)
21	0.585	0.474	NFD-LFPC 5 (Mean)
22	0.582	0.463	NFD-LFPC 12 (25th percentile)
23	0.574	0.492	NFD-LFPC 6 (Mean)
24	0.570	0.523	NTD-LFPC 9 (Mean)
25	0.565	0.514	NTD-LFPC 8 (50th percentile)
26	0.564	0.504	NTD-LFPC 8 (75th percentile)
27	0.560	0.470	NTD-LFPC 3 (75th percentile)
28	0.554	0.480	NTD-LFPC 12 (50th percentile)
29	0.552	0.516	NFD-LFPC 9 (Mean)
30	0.549	0.512	NTD-LFPC 10 (Mean)
31	0.544	0.490	NFD-LFPC 8 (Mean)
32	0.542	0.470	Intensity (50th percentile)
33	0.542	0.502	NTD-LFPC 11 (50th percentile)
34	0.541	0.509	NTD-LFPC 4 (50th percentile)
35	0.538	0.504	NTD-LFPC 9 (50th percentile)

Continued on next page

Table A.1—continued from previous page

Rank	IG	CC_P	Feature
36	0.538	0.495	NFD-LFPC 11 (50th percentile)
37	0.535	0.497	NFD-LFPC 10 (Mean)
38	0.529	0.480	NTD-LFPC 6 (50th percentile)
39	0.528	0.490	NTD-LFPC 10 (75th percentile)
40	0.522	0.448	NFD-LFPC 12 (50th percentile)
41	0.521	0.410	NFD-LFPC 8 (25th percentile)
42	0.521	0.466	Intensity (25th percentile)
43	0.521	0.479	NTD-LFPC 3 (50th percentile)
44	0.521	0.499	NTD-LFPC 11 (25th percentile)
45	0.520	0.428	NFD-LFPC 12 (Mean)
46	0.518	0.444	NTD-LFPC 12 (75th percentile)
47	0.515	0.469	NTD-LFPC 12 (Mean)
48	0.514	0.443	NTD-LFPC 1 (25th percentile)
49	0.513	0.503	NTD-LFPC 10 (50th percentile)
50	0.508	0.461	NTD-LFPC 4 (25th percentile)
51	0.501	0.476	NFD-LFPC 6 (75th percentile)
52	0.499	0.485	NTD-LFPC 6 (75th percentile)
53	0.499	0.463	Intensity (Mean)
54	0.498	0.454	NFD-LFPC 8 (75th percentile)
55	0.497	0.504	NTD-LFPC 2 (75th percentile)
56	0.497	0.501	NFD-LFPC 10 (25th percentile)
57	0.496	0.495	NFD-LFPC 9 (75th percentile)
58	0.489	0.468	NFD-LFPC 10 (50th percentile)
59	0.487	0.480	NFD-LFPC 11 (75th percentile)
60	0.486	0.483	NTD-LFPC 9 (25th percentile)
61	0.480	0.450	NFD-LFPC 6 (50th percentile)
62	0.479	0.486	NTD-LFPC 10 (25th percentile)
63	0.478	0.432	Utterance duration
64	0.477	0.456	NTD-LFPC 12 (25th percentile)
65	0.475	0.482	NFD-LFPC 9 (50th percentile)
66	0.472	0.478	NTD-LFPC 2 (25th percentile)
67	0.465	0.462	Intensity (Mean − Min)
68	0.464	0.456	NFD-LFPC 2 (50th percentile)
69	0.464	0.480	NFD-LFPC 9 (25th percentile)
70	0.460	0.440	NFD-LFPC 8 (50th percentile)
71	0.459	0.481	NTD-LFPC 1 (75th percentile)
72	0.458	0.456	NTD-LFPC 8 (25th percentile)
73	0.456	0.449	NFD-LFPC 2 (75th percentile)
74	0.453	0.447	NTD-LFPC 6 (25th percentile)
75	0.453	0.421	Phoneme speaking rate

Continued on next page

Table A.1—continued from previous page

Rank	IG	CC_P	Feature
76	0.452	0.466	NFD-LFPC 7 (Mean)
77	0.445	0.429	Syllable speaking rate
78	0.445	0.434	NFD-LFPC 6 (25th percentile)
79	0.438	0.452	Intensity (75th percentile)
80	0.438	0.453	NFD-LFPC 2 (Mean)
81	0.436	0.469	NFD-LFPC 10 (75th percentile)
82	0.436	0.445	NTD-LFPC 5 (25th percentile)
83	0.434	0.420	NFD-LFPC 7 (25th percentile)
84	0.434	0.442	NFD-LFPC 5 (50th percentile)
85	0.431	0.467	NFD-LFPC 5 (75th percentile)
86	0.426	0.423	NFD-LFPC 5 (25th percentile)
87	0.423	0.429	NTD-LFPC 3 (25th percentile)
88	0.420	0.451	NFD-LFPC 4 (Mean)
89	0.401	0.451	NFD-LFPC 7 (75th percentile)
90	0.401	0.420	NFD-LFPC 4 (75th percentile)
91	0.391	0.401	Syllable speaking rate (Mean − Min)
92	0.391	0.413	NFD-LFPC 4 (50th percentile)
93	0.369	0.430	NTD-LFPC 7 (25th percentile)
94	0.366	0.395	Silent pause duration
95	0.365	0.380	NFD-LFPC 12 (75th percentile)
96	0.364	0.421	NFD-LFPC 7 (50th percentile)
97	0.359	0.400	NFD-LFPC 8 (Max)
98	0.358	0.390	Spectral centroid (75th percentile)
99	0.355	0.434	NFD-LFPC 4 (25th percentile)
100	0.351	0.375	Silent pause frequency

Abbreviations: IG (information gain), CC_P (Pearson's correlation coefficient)

Table A.2 Top 100 features ranked by information gain for CLSE-Dual

Rank	IG	CC_P	Feature
1	0.620	0.470	Silent pause frequency ratio
2	0.374	0.429	Silent pause frequency
3	0.331	0.357	Intensity (Mean − Min)
4	0.323	0.399	NTD-LFPC 3 (50th percentile)
5	0.320	0.339	Phoneme speaking rate (SD)
6	0.316	0.366	NTD-LFPC 6 (50th percentile)
7	0.307	0.370	NTD-LFPC 4 (Mean)
8	0.302	0.372	NTD-LFPC 5 (50th percentile)
9	0.302	0.379	NTD-LFPC 3 (Mean)
10	0.292	0.373	NTD-LFPC 4 (50th percentile)
11	0.282	0.386	NTD-LFPC 2 (50th percentile)
12	0.279	0.372	NTD-LFPC 1 (Mean)
13	0.278	0.336	NTD-LFPC 10 (75th percentile)
14	0.273	0.376	NTD-LFPC 2 (Mean)
15	0.273	0.347	NTD-LFPC 10 (Mean)
16	0.270	0.340	NTD-LFPC 10 (50th percentile)
17	0.270	0.319	NTD-LFPC 12 (Mean)
18	0.268	0.324	NFD-LFPC 11 (50th percentile)
19	0.268	0.320	NFD-LFPC 10 (75th percentile)
20	0.266	0.336	NTD-LFPC 11 (50th percentile)
21	0.266	0.321	NFD-LFPC 10 (Mean)
22	0.262	0.354	NTD-LFPC 8 (50th percentile)
23	0.260	0.341	NFD-LFPC 7 (25th percentile)
24	0.259	0.348	Intensity (25th percentile)
25	0.258	0.303	Syllable duration (Max)
26	0.258	0.348	Intensity (Mean)
27	0.257	0.319	NTD-LFPC 12 (75th percentile)
28	0.256	0.335	Intensity (75th percentile)
29	0.256	0.359	NFD-LFPC 7 (50th percentile)
30	0.256	0.378	NFD-LFPC 4 (Mean)
31	0.255	0.362	NTD-LFPC 8 (Mean)
32	0.252	0.359	NTD-LFPC 7 (75th percentile)
33	0.252	0.356	NTD-LFPC 9 (Mean)
34	0.252	0.325	NTD-LFPC 11 (75th percentile)
35	0.250	0.348	NTD-LFPC 1 (50th percentile)
36	0.250	0.291	NFD-LFPC 12 (75th percentile)
37	0.250	0.350	NTD-LFPC 8 (75th percentile)
38	0.248	0.339	NTD-LFPC 9 (50th percentile)
39	0.248	0.331	NTD-LFPC 11 (Mean)
40	0.244	0.281	NFD-LFPC 12 (Mean)

Continued on next page

Table A.2—continued from previous page

Rank	IG	CC_P	Feature
41	0.244	0.303	NTD-LFPC 9 (75th percentile)
42	0.243	0.357	NTD-LFPC 3 (25th percentile)
43	0.241	0.360	NTD-LFPC 9 (25th percentile)
44	0.240	0.361	NTD-LFPC 7 (50th percentile)
45	0.238	0.358	NTD-LFPC 6 (Mean)
46	0.238	0.340	NTD-LFPC 8 (25th percentile)
47	0.237	0.338	NTD-LFPC 4 (25th percentile)
48	0.237	0.337	Intensity (50th percentile)
49	0.236	0.350	NFD-LFPC 4 (75th percentile)
50	0.236	0.321	NFD-LFPC 3 (Mean)
51	0.236	0.291	Syllable duration (Max − Mean)
52	0.236	0.356	NTD-LFPC 7 (Mean)
53	0.235	0.326	ΔIntensity (SD)
54	0.235	0.352	NFD-LFPC 4 (50th percentile)
55	0.233	0.352	NTD-LFPC 5 (Mean)
56	0.233	0.328	NTD-LFPC 3 (75th percentile)
57	0.232	0.306	NTD-LFPC 12 (50th percentile)
58	0.231	0.240	TEO-CB-Auto-Env 15 (Mean − Min)
59	0.230	0.334	NTD-LFPC 11 (Mean − Min)
60	0.230	0.350	NFD-LFPC 7 (Mean)
61	0.226	0.350	NTD-LFPC 1 (75th percentile)
62	0.225	0.322	NTD-LFPC 4 (75th percentile)
63	0.223	0.347	NTD-LFPC 5 (25th percentile)
64	0.223	0.327	NTD-LFPC 2 (75th percentile)
65	0.223	0.288	ΔNTD-LFPC 2 (50th percentile)
66	0.220	0.312	NFD-LFPC 9 (75th percentile)
67	0.217	0.254	ΔNTD-LFPC 3 (50th percentile)
68	0.217	0.336	NFD-LFPC 8 (75th percentile)
69	0.217	0.323	NFD-LFPC 10 (50th percentile)
70	0.216	0.359	NTD-LFPC 6 (25th percentile)
71	0.215	0.349	NTD-LFPC 10 (25th percentile)
72	0.214	0.339	NTD-LFPC 7 (25th percentile)
73	0.214	0.288	ΔNTD-LFPC 4 (75th percentile)
74	0.212	0.335	NFD-LFPC 6 (Mean)
75	0.211	0.313	NFD-LFPC 5 (Mean)
76	0.208	0.281	Speech segment length (Max)
77	0.208	0.332	NTD-LFPC 2 (25th percentile)
78	0.208	0.321	NFD-LFPC 3 (75th percentile)
79	0.208	0.314	NFD-LFPC 3 (50th percentile)
80	0.207	0.319	NFD-LFPC 9 (50th percentile)

Continued on next page

Table A.2—continued from previous page

Rank	IG	CC_P	Feature
81	0.206	0.271	NFD-LFPC 12 (50th percentile)
82	0.206	0.298	NFD-LFPC 11 (Mean)
83	0.205	0.328	NFD-LFPC 4 (25th percentile)
84	0.205	0.322	NTD-LFPC 6 (75th percentile)
85	0.203	0.329	NTD-LFPC 1 (25th percentile)
86	0.203	0.312	NTD-LFPC 11 (25th percentile)
87	0.199	0.327	NFD-LFPC 9 (Mean)
88	0.197	0.285	NFD-LFPC 4 (Max)
89	0.196	0.326	NFD-LFPC 10 (25th percentile)
90	0.194	0.311	NFD-LFPC 10 (Mean − Min)
91	0.194	0.338	Mean silent pause duration
92	0.192	0.248	TEO-CB-Auto-Env 14 (Max)
93	0.192	0.311	Phoneme speaking rate
94	0.191	0.246	NTD-LFPC 8 (Regr. intercept)
95	0.190	0.305	NTD-LFPC 10 (Mean − Min)
96	0.190	0.303	NFD-LFPC 11 (75th percentile)
97	0.189	0.280	NTD-LFPC 7 (Position of max)
98	0.188	0.325	NFD-LFPC 8 (50th percentile)
99	0.187	0.278	Speech segment length (SD)
100	0.186	0.321	NFD-LFPC 6 (50th percentile)

Abbreviations: IG (information gain), CC_P (Pearson's correlation coefficient)

Table A.3 Top 100 features ranked by information gain for CLSE-Span

Rank	IG	CC_P	Feature
1	0.338	0.398	Onset latency
2	0.060	0.127	Intensity (Position of max)
3	0.058	0.177	F_0 (Regr. intercept)
4	0.057	0.104	F_2 centre (Regr. slope)
5	0.057	0.101	NTD-LFPC 3 (Position of max)
6	0.047	0.125	NFD-LFPC 3 (Position of max)
7	0.047	0.122	NTD-LFPC 9 (Position of max)
8	0.044	0.085	NTD-LFPC 5 (Position of max)
9	0.043	0.089	Intensity (Position of max)
10	0.043	0.089	NTD-LFPC 6 (Position of max)
11	0.041	0.090	Intensity (Position of min)
12	0.039	0.097	NFD-LFPC 5 (Position of max)
13	0.038	0.070	ΔΔIntensity (Position of min)
14	0.038	0.098	NFD-LFPC 1 (Position of max)
15	0.037	0.119	NTD-LFPC 11 (Position of max)
16	0.036	0.144	MFCC 1 (75th percentile)
17	0.035	0.080	NTD-LFPC 4 (Position of max)
18	0.035	0.089	NTD-LFPC 8 (Position of max)
19	0.034	0.070	NFD-LFPC 3 (Position of max)
20	0.034	0.086	ΔΔNTD-LFPC 5 (Position of min)
21	0.034	0.091	NTD-LFPC 12 (Position of max)
22	0.034	0.081	TEO-CB-Auto-Env 13 (Position of max)
23	0.033	0.048	TEO-CB-Auto-Env 11 (Position of max)
24	0.033	0.112	NTD-LFPC 10 (Position of max)
25	0.032	0.096	TEO-CB-Auto-Env 16 (Position of max)
26	0.032	0.095	NTD-LFPC 7 (Position of max)
27	0.032	0.105	MFCC 3 (Position of max)
28	0.031	0.065	NTD-LFPC 12 (Position of min)
29	0.030	0.085	NTD-LFPC 2 (Position of max)
30	0.030	0.068	TEO-CB-Auto-Env 8 (Position of max)
31	0.030	0.128	MFCC 1 (Max)
32	0.030	0.117	MFCC 6 (75th percentile)
33	0.029	0.079	NFD-LFPC 4 (Position of max)
34	0.029	0.054	NFD-LFPC 12 (Position of min)
35	0.029	0.088	NFD-LFPC 9 (Position of max)
36	0.029	0.093	NFD-LFPC 7 (Position of max)
37	0.029	0.071	ΔΔNFD-LFPC 10 (Position of min)
38	0.028	0.076	NFD-LFPC 6 (Position of max)
39	0.028	0.057	NTD-LFPC 10 (Position of min)
40	0.028	0.094	TEO-CB-Auto-Env 14 (Position of min)

Continued on next page

Table A.3—continued from previous page

Rank	IG	CC_P	Feature
41	0.028	0.057	NTD-LFPC 11 (Position of min)
42	0.028	0.099	NFD-LFPC 10 (Position of max)
43	0.028	0.098	TEO-CB-Auto-Env 10 (Position of max)
44	0.027	0.074	NFD-LFPC 8 (Position of max)
45	0.027	0.052	NTD-LFPC 5 (Position of min)
46	0.027	0.072	NTD-LFPC 1 (Position of min)
47	0.026	0.132	MFCC 1 (SD)
48	0.026	0.062	NTD-LFPC 7 (Position of min)
49	0.026	0.079	NTD-LFPC 1 (Position of max)
50	0.026	0.083	ΔΔIntensity (Position of max)
51	0.026	0.042	F_1 bandwidth (Position of min)
52	0.026	0.069	NTD-LFPC 9 (Position of min)
53	0.026	0.072	ΔΔNFD-LFPC 12 (Position of min)
54	0.025	0.112	MFCC 3 (Regr. intercept)
55	0.025	0.134	NFD-LFPC 12 (25th percentile)
56	0.025	0.053	NFD-LFPC 8 (Position of min)
57	0.025	0.087	TEO-CB-Auto-Env 14 (Position of max)
58	0.025	0.086	CPP (Position of min)
59	0.025	0.088	NFD-LFPC 9 (50th percentile)
60	0.025	0.086	HNR (Position of max)
61	0.024	0.035	ΔΔMFCC 5 (Position of min)
62	0.024	0.121	CPP (Regr. intercept)
63	0.024	0.074	TEO-CB-Auto-Env 14 (Position of max)
64	0.024	0.054	TEO-CB-Auto-Env 3 (Position of max)
65	0.024	0.089	ΔΔTEO-CB-Auto-Env 16 (Position of max)
66	0.023	0.059	NFD-LFPC 1 (Regr. intercept)
67	0.023	0.086	CPP (Position of max)
68	0.023	0.035	NFD-LFPC 10 (Position of max)
69	0.023	0.105	F_2 centre (Regr. intercept)
70	0.023	0.074	NFD-LFPC 11 (Position of max)
71	0.023	0.098	ΔΔNTD-LFPC 3 (Position of min)
72	0.023	0.058	ΔΔNTD-LFPC 12 (Position of max)
73	0.023	0.072	TEO-CB-Auto-Env 9 (Position of max)
74	0.022	0.071	TEO-CB-Auto-Env 4 (Position of max)
75	0.022	0.080	ΔΔTEO-CB-Auto-Env 14 (Position of min)
76	0.022	0.090	Jitter (Position of min)
77	0.022	0.087	ΔΔCPP (Position of max)
78	0.022	0.075	ΔΔNTD-LFPC 12 (Position of min)
79	0.022	0.091	ΔΔNTD-LFPC 11 (Position of min)
80	0.022	0.091	MFCC 3 (Regr. slope)

Continued on next page

Table A.3—continued from previous page

Rank	IG	CC_P	Feature
81	0.022	0.052	NTD-LFPC 8 (Position of max)
82	0.022	0.064	ΔΔNTD-LFPC 10 (Position of min)
83	0.022	0.085	TEO-CB-Auto-Env 3 (Position of max)
84	0.022	0.053	NTD-LFPC 6 (Position of min)
85	0.022	0.066	F_2 centre (Min)
86	0.022	0.061	NTD-LFPC 3 (Position of min)
87	0.022	0.094	NFD-LFPC 2 (Position of max)
88	0.022	0.094	ΔΔTEO-CB-Auto-Env 15 (Position of max)
89	0.022	0.065	ΔΔNTD-LFPC 4 (Position of min)
90	0.022	0.059	MFCC 7 (Regr. intercept)
91	0.022	0.068	MFCC 2 (Regr. intercept)
92	0.021	0.064	NTD-LFPC 12 (Position of max)
93	0.021	0.059	ΔΔNFD-LFPC 5 (Position of max)
94	0.021	0.080	NFD-LFPC 9 (Position of min)
95	0.021	0.056	NFD-LFPC 9 (Position of max)
96	0.021	0.097	ΔΔNTD-LFPC 6 (Position of min)
97	0.021	0.101	MFCC 7 (25th percentile)
98	0.021	0.043	F_0 (Regr. slope)
99	0.021	0.068	NTD-LFPC 7 (Position of max)
100	0.021	0.094	ΔΔNTD-LFPC 2 (Position of min)

Abbreviations: IG (information gain), CC_P (Pearson's correlation coefficient)

Table A.4 Top 100 features ranked by information gain for CoLoSS

Rank	IG	CC_P	Feature
1	0.549	0.223	Mean silent pause duration
2	0.549	0.215	Silent pause frequency ratio
3	0.515	0.223	Silent pause duration ratio
4	0.488	0.225	Silent pause duration
5	0.458	0.220	Silent pause frequency
6	0.455	0.065	Syllable speaking rate (Min)
7	0.375	0.016	Syllable speaking rate (Max)
8	0.267	0.064	Phoneme speaking rate (Max)
9	0.173	0.211	Phoneme speaking rate (Min)
10	0.154	0.234	Utterance duration
11	0.080	0.204	Phoneme speaking rate
12	0.042	0.040	Phoneme speaking rate (Position of min)
13	0.042	0.043	NTD-LFPC 11 (Position of min)
14	0.042	0.048	Phoneme duration (Position of max)
15	0.041	0.022	NTD-LFPC 12 (Position of min)
16	0.036	0.021	NTD-LFPC 3 (Position of min)
17	0.035	0.036	Syllable speaking rate (Regr. slope)
18	0.034	0.044	NTD-LFPC 5 (Position of min)
19	0.031	0.036	NFD-LFPC 12 (Position of min)
20	0.030	0.124	Phoneme articulation rate
21	0.030	0.008	Syllable speaking rate (Regr. intercept)
22	0.029	0.023	Syllable speaking rate (Mean − Min)
23	0.027	0.147	Phoneme speaking rate (SD)
24	0.026	0.123	Phoneme duration (Mean)
25	0.026	0.122	Intensity (Regr. slope)
26	0.024	0.110	Phoneme duration (Mean − Min)
27	0.024	0.124	Phoneme speaking rate (Mean − Min)
28	0.024	0.090	Syllable speaking rate
29	0.023	0.140	Phoneme speaking rate (Max − Mean)
30	0.023	0.023	Syllable speaking rate (Position of max)
31	0.023	0.137	Phoneme duration (SD)
32	0.023	0.029	NFD-LFPC 11 (Position of min)
33	0.022	0.091	NTD-LFPC 9 (Regr. slope)
34	0.021	0.104	Consonant duration (Mean)
35	0.021	0.100	NTD-LFPC 11 (Regr. slope)
36	0.020	0.107	Consonant duration (Max)
37	0.020	0.101	NTD-LFPC 3 (Regr. slope)
38	0.019	0.091	NTD-LFPC 10 (Regr. slope)
39	0.019	0.110	Phoneme duration (Max)
40	0.019	0.106	NTD-LFPC 12 (Regr. slope)

Continued on next page

Table A.4—continued from previous page

Rank	IG	CC_P	Feature
41	0.019	0.076	NTD-LFPC 7 (Regr. slope)
42	0.019	0.055	ΔΔNTD-LFPC 7 (SD)
43	0.018	0.085	NTD-LFPC 8 (Regr. slope)
44	0.018	0.095	Phoneme speaking rate (Regr. intercept)
45	0.018	0.079	ΔΔMFCC 1 (25th percentile)
46	0.017	0.060	ΔΔNTD-LFPC 9 (SD)
47	0.017	0.034	Syllable duration (Max)
48	0.017	0.058	ΔΔNTD-LFPC 6 (SD)
49	0.017	0.113	Consonant duration (SD)
50	0.017	0.051	ΔNFD-LFPC 2 (SD)
51	0.017	0.055	ΔΔNTD-LFPC 10 (SD)
52	0.017	0.104	Consonant duration (Mean − Min)
53	0.017	0.072	ΔΔNTD-LFPC 7 (75th percentile)
54	0.017	0.054	ΔΔNTD-LFPC 8 (SD)
55	0.017	0.073	NTD-LFPC 6 (Regr. slope)
56	0.016	0.077	ΔΔNTD-LFPC 6 (75th percentile)
57	0.016	0.078	NFD-LFPC 9 (Regr. slope)
58	0.016	0.084	NFD-LFPC 11 (Regr. slope)
59	0.016	0.021	Syllable articulation rate
60	0.016	0.050	ΔNFD-LFPC 4 (SD)
61	0.016	0.069	ΔΔNTD-LFPC 11 (75th percentile)
62	0.016	0.031	ΔNTD-LFPC 11 (Regr. slope)
63	0.016	0.092	Consonant duration (Max − Mean)
64	0.015	0.073	NTD-LFPC 5 (Regr. slope)
65	0.015	0.066	ΔΔSpectral centroid (75th percentile)
66	0.015	0.043	ΔΔNFD-LFPC 5 (SD)
67	0.015	0.070	NFD-LFPC 12 (Regr. slope)
68	0.015	0.017	Syllable duration (Mean)
69	0.015	0.059	ΔΔNTD-LFPC 5 (SD)
70	0.015	0.074	ΔΔMFCC 1 (SD)
71	0.015	0.068	ΔΔNTD-LFPC 4 (SD)
72	0.015	0.095	ΔΔMFCC 2 (75th percentile)
73	0.015	0.038	ΔNFD-LFPC 5 (SD)
74	0.015	0.079	NTD-LFPC 4 (Regr. slope)
75	0.015	0.038	ΔNTD-LFPC 12 (Regr. slope)
76	0.014	0.072	ΔΔMFCC 3 (SD)
77	0.014	0.026	ΔΔNFD-LFPC 2 (SD)
78	0.014	0.067	ΔΔNTD-LFPC 10 (75th percentile)
79	0.014	0.041	ΔΔNTD-LFPC 11 (SD)
80	0.014	0.044	ΔNTD-LFPC 5 (75th percentile)

Continued on next page

Table A.4—continued from previous page

Rank	IG	CC_P	Feature
81	0.014	0.079	ΔΔNFD-LFPC 12 (75th percentile)
82	0.014	0.062	ΔΔNTD-LFPC 2 (SD)
83	0.014	0.057	ΔΔNTD-LFPC 3 (SD)
84	0.014	0.008	ΔNFD-LFPC 5 (Mean)
85	0.014	0.079	F_0 (Regr. slope)
86	0.014	0.075	ΔΔNFD-LFPC 6 (75th percentile)
87	0.014	0.088	ΔΔNFD-LFPC 10 (75th percentile)
88	0.014	0.042	ΔNTD-LFPC 6 (SD)
89	0.014	0.040	ΔNFD-LFPC 3 (Regr. slope)
90	0.014	0.091	Phoneme duration (Max − Mean)
91	0.014	0.039	ΔΔIntensity (Regr. slope)
92	0.014	0.022	NTD-LFPC 7 (Position of min)
93	0.013	0.061	NFD-LFPC 5 (Regr. slope)
94	0.013	0.074	ΔΔNFD-LFPC 8 (75th percentile)
95	0.013	0.060	NFD-LFPC 10 (Regr. slope)
96	0.013	0.018	ΔΔNFD-LFPC 3 (SD)
97	0.013	0.046	ΔNTD-LFPC 6 (75th percentile)
98	0.013	0.059	NTD-LFPC 1 (Regr. slope)
99	0.013	0.014	ΔNFD-LFPC 5 (75th percentile)
100	0.013	0.053	ΔΔNFD-LFPC 7 (SD)

Abbreviations: IG (information gain), CC_P (Pearson's correlation coefficient)

Appendix B

Hyperparameter Settings

Appendix B.1 to B.5 present the optimal model-hyperparameter settings from the experiments conducted in this thesis (Section 6.3 to 6.7).

B.1 Within-Corpus Evaluation

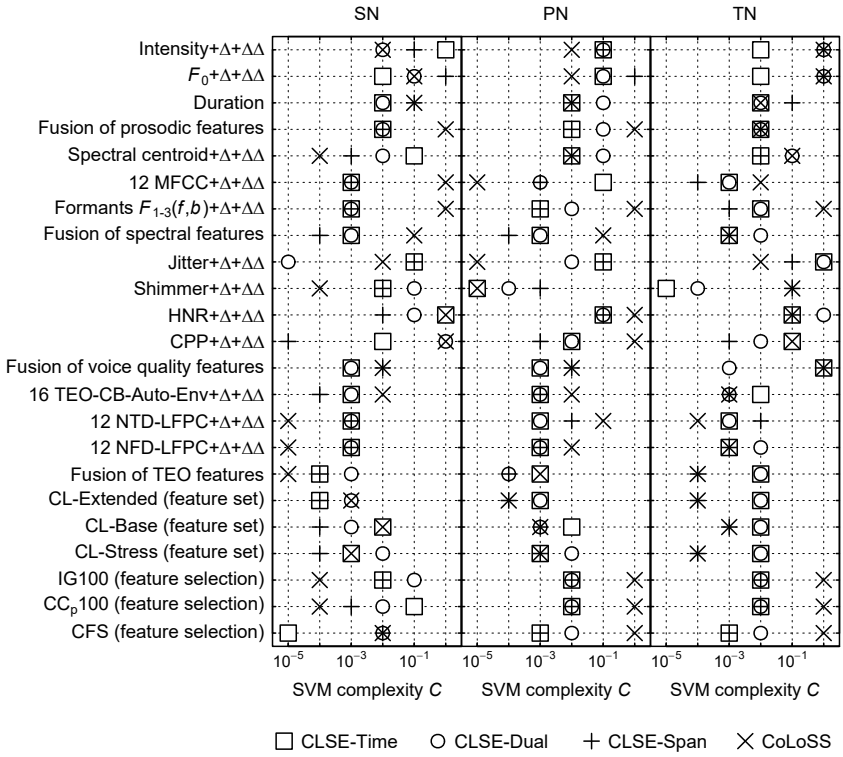


Figure B.1 Optimal SVM complexity C per feature category (vertically arranged) and normalisation method (horizontally arranged, above) for within-corpus evaluation. Symbols indicate corpora. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation)

B.2 Cross-Corpus Evaluation

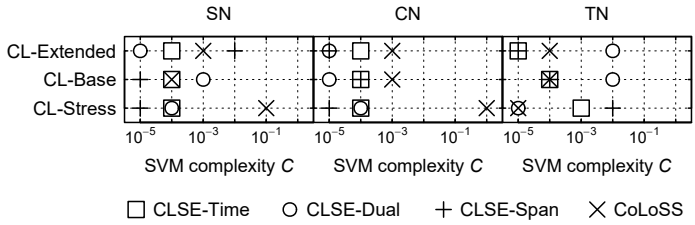


Figure B.2 Optimal SVM complexity C per feature set (vertically arranged) and feature normalisation method (horizontally arranged, above) for cross-corpus evaluation by training on one corpus and testing against the remaining corpora. Symbols indicate corpora used for training. Abbreviations: CN (corpus normalisation), SN (speaker normalisation), TN (training normalisation)

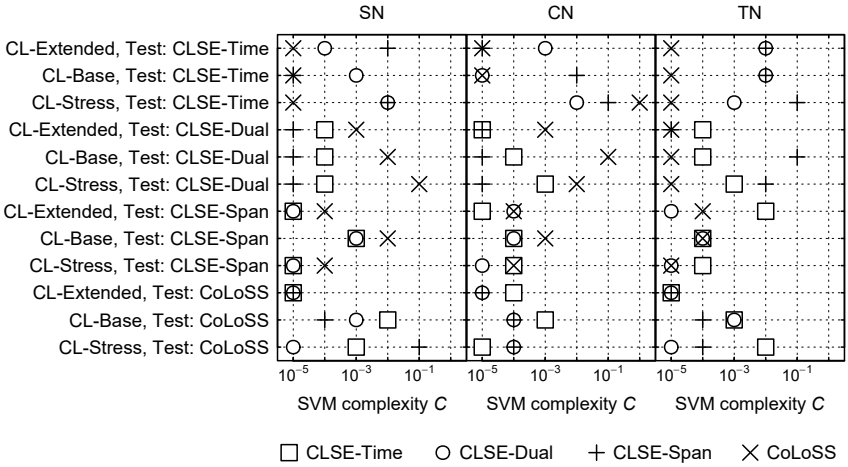


Figure B.3 Optimal SVM complexity C per feature set and individual test case (vertically arranged), and feature normalisation method (horizontally arranged, above) for cross-corpus evaluation. Symbols indicate corpora used for training. Abbreviations: CN (corpus normalisation), SN (speaker normalisation), TN (training normalisation)

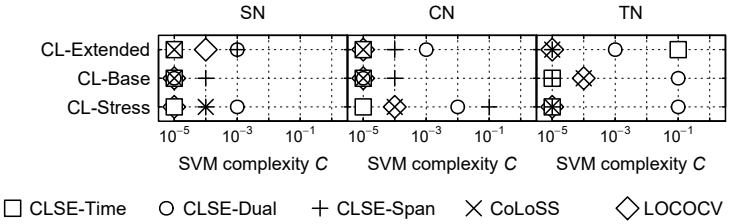


Figure B.4 Optimal SVM complexity C per feature set (vertically arranged) and feature normalisation method (horizontally arranged, above) for cross-corpus evaluation by corpus fusion for model training and testing against a separate corpus. Symbols indicate corpora used for testing and LOCOCV (leave-one-corpus-out cross-validation with constant C). Abbreviations: CN (corpus normalisation), SN (speaker normalisation), TN (training normalisation)

B.3 Mixed-Corpus Evaluation

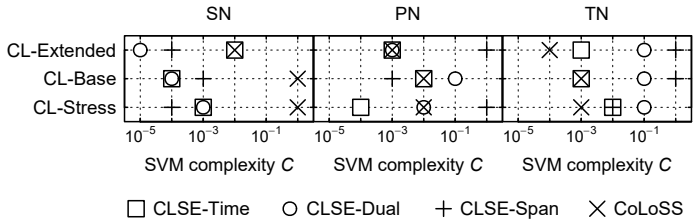


Figure B.5 Optimal SVM complexity C per feature set (vertically arranged) and feature normalisation method (horizontally arranged, above) for mixed-corpus evaluation. Symbols indicate corpora used for testing. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation)

B.4 COMPARE 2014—Cognitive Load Sub-Challenge

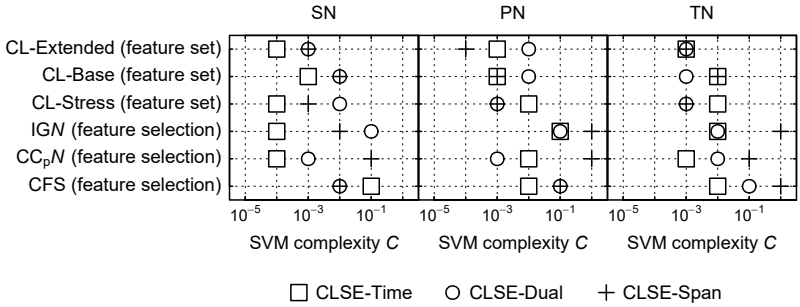


Figure B.6 SVM complexity C optimised on the development set of the Cognitive Load Sub-Challenge per feature category (vertically arranged) and feature normalisation method (horizontally arranged, above). Symbols indicate corpora. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation)

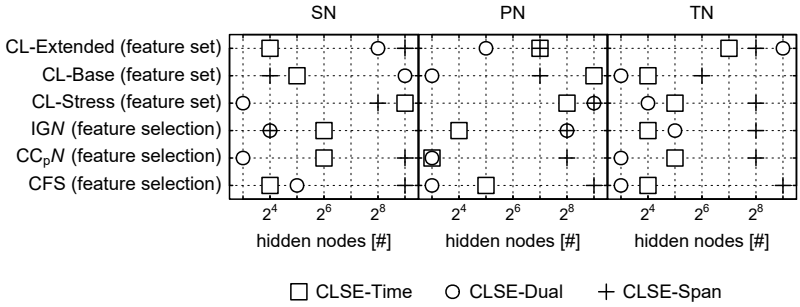


Figure B.7 Number of MLP hidden nodes optimised on the development set of the Cognitive Load Sub-Challenge per feature category (vertically arranged) and feature normalisation method (horizontally arranged, above). Symbols indicate corpora. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation)

B.5 Regression Approaches

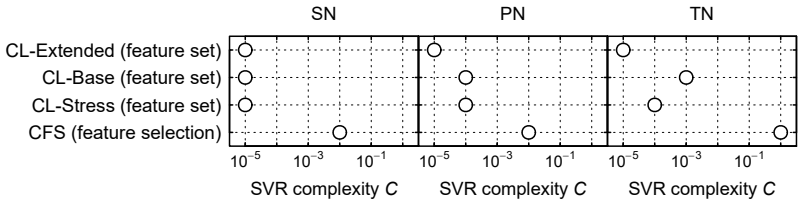


Figure B.8 Optimal SVR complexity C per feature category (vertically arranged) and feature normalisation method (horizontally arranged, above) for the prediction of the secondary task efficiency of CoLoSS. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation)

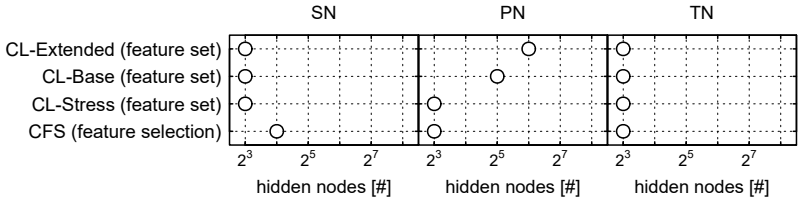


Figure B.9 Optimal number of MLP hidden nodes per feature category (vertically arranged) and feature normalisation method (horizontally arranged, above) for the prediction of the secondary task efficiency of CoLoSS. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation)

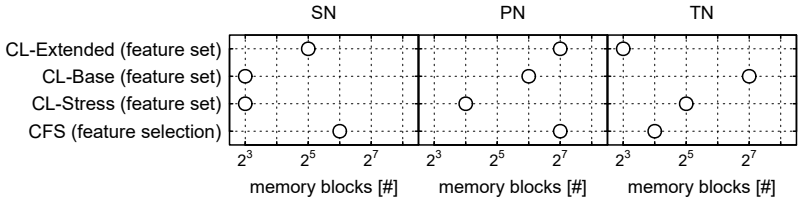


Figure B.10 Optimal number of LSTM memory blocks per feature category (vertically arranged) and feature normalisation method (horizontally arranged, above) for the prediction of the secondary task efficiency of CoLoSS. Abbreviations: PN (partition normalisation), SN (speaker normalisation), TN (training normalisation)

Bibliography

Abhang, P. A., Gawali, B. W., and Mehrotra, S. C. (2016). *Introduction to EEG- and Speech-Based Emotion Recognition*. Academic Press.

Anagnostopoulos, C.-N., Iliou, T., and Giannoukos, I. (2015). Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011. *Artificial Intelligence Review*, 43(2):155–177.

Antonenko, P. D. and Niederhauser, D. S. (2010). The influence of leads on cognitive load and learning in a hypertext environment. *Computers in Human Behavior*, 26(2):140–150.

Anvari, F., Tran, H. M. T., and Kavakli, M. (2013). Using Cognitive Load Measurement and Spatial Ability Test to Identify Talented Students in Three-Dimensional Computer Graphics Programming. *International Journal of Information and Education Technology*, 3(1):94–99.

Ark, W. S., Dryer, D. C., and Lu, D. J. (1999). The Emotion Mouse. In *Proceedings of the 8th International Conference on Human-Computer Interaction (HCI International 1999)*, pp. 818–823, Hillsdale, NJ, USA. ACM.

Atkinson, R. C. and Shiffrin, R. M. (1968). Human Memory: A Proposed System and its Control Processes. *Psychology of Learning and Motivation*, 2:89–195.

Audacity Team (2012). Audacity: Free Audio Editor and Recorder. Version 2.0.2. Retrieved from https://sourceforge.net/projects/audacity/. Accessed 23 August 2015.

Backs, R. W. and Walrath, L. C. (1992). Eye movement and pupillary response indices of mental workload during visual search of symbolic displays. *Applied Ergonomics*, 23(4):243–254.

Baddeley, A. (1992). Working Memory. *Science*, 255(5044):556–559.

Baddeley, A. (1996). The fractionation of working memory. *Proceedings of the National Academy of Sciences*, 93(24):13468–13472.

Baddeley, A. (2000). The Episodic Buffer: A New Component of Working Memory? *Trends in Cognitive Sciences*, 4(11):417–423.

Baddeley, A. (2002). Is Working Memory Still Working? *European Psychologist*, 7(2):85–97.

Baddeley, A. (2003). Working Memory and Language: An Overview. *Journal of Communication Disorders*, 36(3):189–208.

Baddeley, A. and Hitch, G. (1974). Working Memory. *Psychology of Learning and Motivation*, 8:47–89.

Bartlett, F. (1932). *Remembering: A study in experimental and social psychology*. Cambridge University Press.

Berthold, A. and Jameson, A. (1999). Interpreting Symptoms of Cognitive Load in Speech Input. In *Proceedings of the Seventh International Conference on User Modeling (UM 99)*, pp. 235–244, Vienna, Austria. Springer.

Berthold, M. R., Borgelt, C., Höppner, F., and Klawonn, F. (2010). *Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data*. Springer Science & Business Media.

Bhatia, S., Hayat, M., and Goecke, R. (2017). A Multimodal System to Characterise Melancholia: Cascaded Bag of Words Approach. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI 2017)*, pp. 274–280, Glasgow, UK. ACM.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer-Verlag New York.

Black, K. (2012). *Business Statistics: For Contemporary Decision Making*. John Wiley & Sons.

Boersma, P. (1993). Accurate Short-term Analysis of the Fundamental Frequency and the Harmonics-to-noise Ratio of a Sampled Sound. In *Proceedings of the Institute of Phonetic Sciences*, volume 17, pp. 97–110.

Boersma, P. (2002). Praat, a system for doing phonetics by computer. *Glot International*, 5:341–345.

Boril, H., Omid Sadjadi, S., Kleinschmidt, T., and Hansen, J. H. (2010). Analysis and Detection of Cognitive Load and Frustration in Drivers' Speech. In *Proceedings of INTERSPEECH 2010*, pp. 502–505, Makuhari, Japan. ISCA.

Brown, J. (1958). Some Tests of the Decay Theory of Immediate Memory. *Quarterly Journal of Experimental Psychology*, 10(1):12–21.

Brunken, R., Plass, J. L., and Leutner, D. (2003). Direct Measurement of Cognitive Load in Multimedia Learning. *Educational Psychologist*, 38(1):53–61.

Brünken, R., Plass, J. L., and Moreno, R. (2010a). Current issues and open questions in cognitive load research. In *Cognitive Load Theory*, pp. 253–272. Cambridge University Press.

Brünken, R., Seufert, T., and Paas, F. (2010b). Measuring cognitive load. In *Cognitive Load Theory*, pp. 181–202. Cambridge University Press.

Campbell, C. and Ying, Y. (2011). *Learning with Support Vector Machines*. Morgan & Claypool Publishers.

Cegarra, J. and Hoc, J.-M. (2006). Cognitive Styles As an Explanation of Experts' Individual Differences: A Case Study in Computer-assisted Troubleshooting Diagnosis. *International Journal of Human-Computer Studies*, 64(2):123–136.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Chen, F., Zhou, J., Wang, Y., Yu, K., Arshad, S. Z., Khawaji, A., and Conway, D. (2016). *Robust Multimodal Cognitive Load Measurement*. Springer.

Chen, L., Mao, X., Xue, Y., and Cheng, L. L. (2012). Speech emotion recognition: Features and classification models. *Digital Signal Processing*, 22(6):1154–1160.

Cierniak, G., Scheiter, K., and Gerjets, P. (2009). Explaining the Split-attention Effect: Is the Reduction of Extraneous Cognitive Load Accompanied by an Increase in Germane Cognitive Load? *Computers in Human Behavior*, 25(2):315–324.

Clark, R. C., Nguyen, F., and Sweller, J. (2011). *Efficiency in Learning: Evidence-Based Guidelines to Manage Cognitive Load*. John Wiley & Sons.

Cortes, C. and Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, 20(3):273–297.

Cover, T. M. and Thomas, J. A. (1991). *Elements of Information Theory*. John Wiley & Sons.

Cowan, N. (1988). Evolving Conceptions of Memory Storage, Selective Attention, and Their Mutual Constraints Within the Human Information-Processing System. *Psychological Bulletin*, 104(2):163–191.

Cowan, N. (1999). An embedded-processes model of working memory. In *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control*, pp. 62–101. Cambridge University Press.

Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, 24(1):87–114.

Coyne, J. T., Baldwin, C., Cole, A., Sibley, C., and Roberts, D. M. (2009). Applying Real Time Physiological Measures of Cognitive Load to Improve Training. In *Proceedings of the International Conference on Foundations of Augmented Cognition (FAC 2009)*, pp. 469–478, San Diego, CA, USA. Springer.

Crundall, D. E. and Underwood, G. (1998). Effects of experience and processing demands on visual information acquisition in drivers. *Ergonomics*, 41(4):448–458.

Cummins, N., Epps, J., Breakspear, M., and Goecke, R. (2011). An Investigation of Depressed Speech Detection: Features and Normalization. In *Proceedings of INTERSPEECH 2011*, pp. 2997–3000, Florence, Italy. ISCA.

Daneman, M. and Carpenter, P. A. (1980). Individual differences in working memory and reading. *Journal of Verbal Learning and Verbal Behavior*, 19(4):450–466.

Davis, S. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366.

De Greef, T., Van Dongen, K., Grootjen, M., and Lindenberg, J. (2007). Augmenting Cognition: Reviewing the Symbiotic Relation Between Man and Machine. In *Proceedings of the International Conference on Foundations of Augmented Cognition (FAC 2007)*, pp. 439–448, Beijing, China. Springer.

De Jong, N. H. and Wempe, T. (2007). Automatic measurement of speech rate in spoken Dutch. In *Proceedings of Amsterdam Center for Language and Communication (ACLC 2007)*, volume 2, pp. 49–58. University of Amsterdam.

De Jong, T. (2010). Cognitive load theory, educational research, and instructional design: Some food for thought. *Instructional science*, 38(2):105–134.

Deeplearning4j Team (2016). Deeplearning4j: Open-source distributed deep learning for the JVM. Apache Software Foundation License. Retrieved from http://deeplearning4j.org/. Accessed 10 July 2017.

Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P., and Ouellet, P. (2011). Front-End Factor Analysis for Speaker Verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798.

Dellaert, F., Polzin, T., and Waibel, A. (1996). Recognizing emotion in speech. In *Proceedings of the Fourth International Conference on Spoken Language Processing (ICSLP 1996)*, pp. 1970–1973, Philadelphia, PA, USA. IEEE.

Dimitriadis, D., Maragos, P., and Potamianos, A. (2005). Auditory Teager Energy Cepstrum Coefficients for Robust Speech Recognition. In *Proceedings of INTERSPEECH 2005*, pp. 3013–3016, Lisbon, Portugal. ISCA.

Droppo, J. and Acero, A. (2008). Environmental Robustness. In *Springer Handbook of Speech Processing*, pp. 653–680. Springer.

Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A. J., and Vapnik, V. (1997). Support Vector Regression Machines. In *Advances in Neural Information Processing Systems 9*, pp. 155–161. MIT Press.

Dubé, A. K. and McEwen, R. N. (2015). Do gestures matter? The implications of using touchscreen devices in mathematics instruction. *Learning and Instruction*, 40:89–98.

Engström, J., Johansson, E., and Östlund, J. (2005). Effects of Visual and Cognitive Load in Real and Simulated Motorway Driving. *Transportation Research Part F: Traffic Psychology and Behaviour*, 8(2):97–120.

Enqing, D., Guizhong, L., Yatong, Z., and Xiaodi, Z. (2002). Applying support vector machines to voice activity detection. In *Proceedings of the 6th International Conference on Signal Processing (ICSP 2002)*, pp. 1124–1127, Beijing, China. IEEE.

Eriksson, A. (2012). Aural/Acoustic vs. Automatic Methods in Forensic Phonetic Case Work. In *Forensic Speaker Recognition*, pp. 41–69. Springer.

Eronen, A. and Klapuri, A. (2000). Musical instrument recognition using cepstral coefficients and temporal features. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2000)*, pp. 753–756, Istanbul, Turkey. IEEE.

Esposito, A., Stejskal, V., Smékal, Z., and Bourbakis, N. (2007). The Significance of Empty Speech Pauses: Cognitive and Algorithmic Issues. In *Proceedings of the International Symposium on Brain, Vision, and Artificial Intelligence (BVAI 2007)*, pp. 542–554, Naples, Italy. Springer.

Eyben, F. (2015). *Real-time Speech and Music Classification by Large Audio Feature Space Extraction*. Springer.

Eyben, F., Weninger, F., Gross, F., and Schuller, B. (2013a). Recent Developments in openSMILE, the Munich Open-source Multimedia Feature Extractor. In *Proceedings of the 21st ACM international conference on Multimedia (ACM MM 2013)*, pp. 835–838, Barcelona, Spain. ACM.

Eyben, F., Weninger, F., Squartini, S., and Schuller, B. (2013b). Real-life voice activity detection with LSTM Recurrent Neural Networks and an application to Hollywood movies. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013)*, pp. 483–487, Vancouver, Canada. IEEE.

Eyben, F., Wöllmer, M., Graves, A., Schuller, B., Douglas-Cowie, E., and Cowie, R. (2010). On-line emotion recognition in a 3-D activation-valence-time continuum using acoustic and linguistic cues. *Journal on Multimodal User Interfaces*, 3(1-2):7–19.

Fan, X., Chen, P.-C., and Yen, J. (2010). Learning HMM-based Cognitive Load Models for Supporting Human-agent Teamwork. *Cognitive Systems Research*, 11(1):108–119.

Fan, X. and Yen, J. (2007). Realistic Cognitive Load Modeling for Enhancing Shared Mental Models in Human-agent Collaboration. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007)*, Honolulu, Hawaii. ACM.

Fant, G. (1960). *Acoustic Theory of Speech Production*. Mouton.

Farinas, J. and Pellegrino, F. (2001). Automatic Rhythm Modeling for Language Identification. In *Proceedings of EUROSPEECH 2001*, pp. 2539–2542, Aalborg, Denmark. ISCA.

Fayyad, U. and Irani, K. (1993). Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pp. 1022–1029, Chambéry, France.

Fellbaum, K. (2012). *Sprachverarbeitung und Sprachübertragung*. Springer.

Fernandez, R. and Picard, R. W. (2003). Modeling drivers' speech under stress. *Speech Communication*, 40(1):145–159.

Field, A. (2013). *Discovering Statistics Using IBM SPSS Statistics*. Sage.

Fletcher, J. (2010). The Prosody of Speech: Timing and Rhythm. In *The Handbook of Phonetic Sciences, Second Edition*, pp. 521–602. Wiley Online Library.

Fraile, R. and Godino-Llorente, J. I. (2014). Cepstral peak prominence: A comprehensive analysis. *Biomedical Signal Processing and Control*, 14:42–54.

Gales, M. and Young, S. (2008). The Application of Hidden Markov Models in Speech Recognition. *Foundations and Trends in Signal Processing*, 1(3):195–304.

Galy, E., Cariou, M., and Mélan, C. (2012). What is the relationship between mental workload factors and cognitive load types? *International Journal of Psychophysiology*, 83(3):269–275.

Gerhard, D. (2003). *Pitch Extraction and Fundamental Frequency: History and Current Techniques*. Department of Computer Science, University of Regina.

Gers, F. A., Schraudolph, N. N., and Schmidhuber, J. (2002). Learning Precise Timing with LSTM Recurrent Networks. *Journal of Machine Learning Research*, 3(Aug):115–143.

Gibbon, D., Moore, R., and Winski, R. (1997). *Handbook of Standards and Resources for Spoken Language Systems*. Walter de Gruyter.

Gikas, J. and Grant, M. M. (2013). Mobile computing devices in higher education: Student perspectives on learning with cellphones, smartphones & social media. *The Internet and Higher Education*, 19:18–26.

Gillmor, S. C., Poggio, J., and Embretson, S. (2015). Effects of Reducing the Cognitive Load of Mathematics Test Items on Student Performance. *Numeracy*, 8(1).

Gorovoy, K., Tung, J., and Poupart, P. (2010). Automatic Speech Feature Extraction for Cognitive Load Classification. In *Proceedings of the Conference of the Canadian Medical and Biological Engineering Society (CMBEC 2010)*, Vancouver, Canada.

Górriz, J. M., Ramírez, J., Lang, E. W., and Puntonet, C. G. (2006). Hard C-means clustering for voice activity detection. *Speech Communication*, 48(12):1638–1649.

Gosztolya, G., Grósz, T., Busa-Fekete, R., and Tóth, L. (2014). Detecting the Intensity of Cognitive and Physical Load Using AdaBoost and Deep Rectifier Neural Networks. In *Proceedings of INTERSPEECH 2014*, pp. 452–456, Singapore. ISCA.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Grootjen, M., Bierman, E., and Neerincx, M. (2006). Optimizing cognitive task load in naval ship control centres: Design of an adaptive interface. In *Proceedings of the 16th World Congress on Ergonomics (IEA 2006)*, Maastricht, Netherlands.

Grootjen, M., Neerincx, M. A., Van Weert, J. C., and Truong, K. P. (2007). Measuring Cognitive Task Load on a Naval Ship: Implications of a Real World Environment. In *Proceedings of the International Conference on Foundations of Augmented Cognition (FAC 2007)*, pp. 147–156, Beijing, China. Springer.

Guo, X., Yin, Y., Dong, C., Yang, G., and Zhou, G. (2008). On the Class Imbalance Problem. In *Proceedings of the Fourth International Conference on Natural Computation (ICNC 2008)*, pp. 192–201, Jinan, China. IEEE.

Gütl, C., Pivec, M., Trummer, C., García-Barrios, V. M., Mödritscher, F., Pripfl, J., and Umgeher, M. (2005). AdeLE (Adaptive e-Learning with Eye-Tracking): Theoretical Background, System Architecture and Application Scenarios. *European Journal of Open, Distance and E-Learning*, 8(2).

Guyon, I. and Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3(Mar):1157–1182.

Hahn, M. and Park, C. K. (1992). An improved speech detection algorithm for isolated Korean utterances. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1992)*, pp. 525–528, San Francisco, CA, USA. IEEE.

Haigh, J. A. and Mason, J. S. (1993). Robust voice activity detection using cepstral features. In *Proceedings of the Region 10 International Conference on Computer, Communication, Control and Power Engineering (TENCON 1993)*, pp. 321–324, Beijing, China. IEEE.

Hall, M. A. (1999). *Correlation-based Feature Selection for Machine Learning*. PhD thesis, University of Waikato, New Zealand.

Harms, K. J. (2013). Applying Cognitive Load Theory to Generate Effective Programming Tutorials. In *Proceedings of the Symposium on Visual Languages and Human Centric Computing*, pp. 179–180, San Jose, CA, USA. IEEE.

Harper, M. P. and Maxwell, M. (2008). Spoken Language Characterization. In *Springer Handbook of Speech Processing*, pp. 797–810. Springer.

Harrington, J. and Cassidy, S. (2012). *Techniques in Speech Acoustics*, volume 8. Springer Science & Business Media.

Hart, S. G. and Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology*, 52:139–183.

Havelock, D., Kuwano, S., and Vorländer, M. (2008). *Handbook of Signal Processing in Acoustics*. Number Bd. 1-2. Springer.

Heman-Ackah, Y. D., Michael, D. D., and Goding, G. S. (2002). The Relationship Between Cepstral Peak Prominence and Selected Parameters of Dysphonia. *Journal of Voice*, 16(1):20–27.

Hermansky, H. (1999). Analysis in Automatic Recognition of Speech. In *Speech Processing, Recognition and Artificial Neural Networks*, pp. 115–137. Springer.

Herms, R. (2016). Prediction of Deception and Sincerity from Speech Using Automatic Phone Recognition-Based Features. In *Proceedings of INTERSPEECH 2016*, pp. 2036–2040, San Francisco, CA, USA. ISCA.

Herms, R., Wirzberger, M., Eibl, M., and Rey, G. D. (2018). CoLoSS: Cognitive Load Corpus with Speech and Performance Data from a Symbol-Digit Dual-Task. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. ELRA.

Hess, A., Jung, J., Maier, A., Taib, R., Yu, K., and Itzstein, B. (2013). Elicitation of mental states and user experience factors in a driving simulator. In *Proceedings of the Intelligent Vehicles Symposium Workshops (IV Workshops 2013)*, pp. 43–48. IEEE.

Hillenbrand, J., Cleveland, R. A., and Erickson, R. L. (1994). Acoustic Correlates of Breathy Vocal Quality. *Journal of Speech, Language, and Hearing Research*, 37(4):769–778.

Hillenbrand, J. and Houde, R. A. (1996). Acoustic Correlates of Breathy Vocal Quality: Dysphonic Voices and Continuous Speech. *Journal of Speech, Language, and Hearing Research*, 39(2):311–321.

Hinkle, D. E., Wiersma, W., and Jurs, S. G. (2003). *Applied Statistics for the Behavioral Sciences*. Houghton Mifflin Boston.

Hochreiter, S., Bengio, Y., Frasconi, P., and Schmidhuber, J. (2001). Gradient Flow in Recurrent Nets: the Difficulty of Learning Long-Term Dependencies. In *Field Guide to Dynamical Recurrent Networks*. IEEE Press.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Hoffman, B. and Schraw, G. (2010). Conceptions of Efficiency: Applications in Learning and Problem Solving. *Educational Psychologist*, 45(1):1–14.

Honda, K. (2008). Physiological Processes of Speech Production. In *Springer Handbook of Speech Processing*, pp. 7–26. Springer.

Honda, M. (2003). Human Speech Production Mechanisms. *NTT Technical Review*, 1(2):24–29.

Hu, P. J.-H., Ma, P.-C., and Chau, P. Y. (1999). Evaluation of user interface designs for information retrieval systems: a computer-based experiment. *Decision support systems*, 27(1-2):125–143.

Huckvale, M. (2014). Prediction of Cognitive Load from Speech with the VOQAL Voice Quality Toolbox for the InterSpeech 2014 Computational Paralinguistics Challenge. In *Proceedings of INTERSPEECH 2014*, pp. 741–745, Singapore. ISCA.

Hunt, A. (1993). Recurrent neural networks for syllabification. *Speech Communication*, 13(3-4):323–332.

Huttunen, K., Keränen, H., Väyrynen, E., Pääkkönen, R., and Leino, T. (2011a). Effect of cognitive load on speech prosody in aviation: Evidence from military simulator flights. *Applied Ergonomics*, 42(2):348–357.

Huttunen, K. H., Keränen, H. I., Pääkkönen, R. J., Päivikki Eskelinen-Rönkä, R., and Leino, T. K. (2011b). Effect of cognitive load on articulation rate and formant frequencies during simulator flights. *The Journal of the Acoustical Society of America*, 129(3):1580–1593.

IPA (1999). *Handbook of the International Phonetic Association: a guide to the use of the International Phonetic Alphabet*. Cambridge University Press.

Izzetoglu, K., Yurtsever, G., Bozkurt, A., Yazici, B., Bunce, S., Pourrezaei, K., and Onaral, B. (2003). NIR spectroscopy measurements of cognitive load elicited by GKT and target categorization. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, Big Island, HI, USA. IEEE.

Jabloun, F. and Cetin, A. E. (1999). The Teager energy based feature parameters for robust speech recognition in car noise. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1999)*, pp. 273–276, Phoenix, AZ, USA. IEEE.

Jacobs, S. C., Friedman, R., Parker, J. D., Tofler, G. H., Jimenez, A. H., Muller, J. E., Benson, H., and Stone, P. H. (1994). Use of skin conductance changes during mental stress testing as an index of autonomic arousal in cardiovascular research. *American Heart Journal*, 128(6):1170–1177.

Jain, A. K., Mao, J., and Mohiuddin, K. M. (1996). Artificial Neural Networks: A Tutorial. *IEEE Computer*, 29(3):31–44.

Jameson, A., Kiefer, J., Müller, C., Großmann-Hutter, B., Wittig, F., and Rummer, R. (2010). Assessment of a User's Time Pressure and Cognitive Load on the Basis of Features of Speech. In *Resource-Adaptive Cognitive Processes*, pp. 171–204. Springer.

Jayan, A. R. (2016). *Speech and Audio Signal Processing*. PHI Learning Pvt. Ltd.

Jing, H., Hu, T.-Y., Lee, H.-S., Chen, W.-C., Lee, C.-C., Tsao, Y., and Wang, H.-M. (2014). Ensemble of Machine Learning Algorithms for Cognitive and Physical Speaker Load Detection. In *Proceedings of INTERSPEECH 2014*, pp. 447–451, Singapore. ISCA.

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning (ECML 1998)*, pp. 137–142, Chemnitz, Germany. Springer.

Johannsen, G. (1979). Workload and workload measurement. In *Mental workload: Its theory and measurement*, pp. 3–11. Springer Science & Business Media.

Jung, J., Maier, A., Gross, A., Ruiz, N., Chen, F., and Yin, B. (2011). Investigating the Effect of Cognitive Load on UX: A Driving Study. In *Proceedings of the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, Salzburg Austria.

Jurafsky, D. and Martin, J. H. (2009). *Speech And Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Pearson.

Just, M. A., Carpenter, P. A., and Miyake, A. (2003). Neuroindices of cognitive workload: Neuroimaging, pupillometric and event-related potential studies of brain work. *Theoretical Issues in Ergonomics Science*, 4(1-2):56–88.

Kaiser, J. F. (1993). Some useful properties of Teager's energy operators. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1993)*, pp. 149–152, Minneapolis, MN, USA. IEEE.

Kalyuga, S. (2011). Cognitive load theory: How many types of load does it really need? *Educational Psychology Review*, 23(1):1–19.

Karthikeyan, P., Murugappan, M., and Yaacob, S. (2014). Analysis of stroop color word test-based human stress detection using electrocardiography and heart rate variability signals. *Arabian Journal for Science and Engineering*, 39(3):1835–1847.

Kassambara, A. (2017). *Machine Learning Essentials: Practical Guide in R*. STHDA.

Keerthi, S. S., Shevade, S. K., Bhattacharyya, C., and Murthy, K. R. K. (2001). Improvements to Platt's SMO algorithm for SVM classifier design. *Neural Computation*, 13(3):637–649.

Kennedy, D. O. and Scholey, A. B. (2000). Glucose administration, heart rate and cognitive performance: effects of increasing mental effort. *Psychopharmacology*, 149(1):63–71.

Keränen, H., Väyrynen, E., Pääkkönen, R., Leino, T., Kuronen, P., Toivanen, J., and Seppänen, T. (2004). Prosodic Features of Speech Produced by Military Pilots During Demanding Tasks. In *Proceedings of the Fonetiikan Paivat Conference*, pp. 88–91.

Khawaja, M. A. (2010). *Cognitive Load Measurement using Speech and Linguistic Features*. PhD thesis, University of New South Wales, Sydney, Australia.

Khawaja, M. A., Chen, F., and Marcus, N. (2012). Analysis of collaborative communication for linguistic cues of cognitive load. *Human Factors*, 54(4):518–529.

Khawaja, M. A., Chen, F., and Marcus, N. (2014). Measuring Cognitive Load Using Linguistic Features: Implications for Usability Evaluation and Adaptive Interaction Design. *International Journal of Human-Computer Interaction*, 30(5):343–368.

Khawaja, M. A., Ruiz, N., and Chen, F. (2007). Potential Speech Features for Cognitive Load Measurement. In *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces (OZCHI 2007)*, pp. 57–60, Adelaide, Australia. ACM.

Khawaja, M. A., Ruiz, N., and Chen, F. (2008). Think Before You Talk: An Empirical Study of Relationship Between Speech Pauses and Cognitive Load. In *Proceedings of the 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat (OZCHI 2008)*, pp. 335–338, Cairns, Australia. ACM.

Kiktova, E. and Juhar, J. (2015). Comparison of Diarization Tools for Building Speaker Database. *Advances in Electrical and Electronic Engineering*, 13(4):314–319.

Kim, H.-G., Moreau, N., and Sikora, T. (2006). *MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval*. John Wiley & Sons.

Kirschner, P. A. (2002). Cognitive Load Theory: Implications of Cognitive Load Theory on the Design of Learning. *Learning and Instruction*, 12(1):1–10.

Kisler, T., Reichel, U., Schiel, F., Draxler, C., Jackl, B., and Pörner, N. (2016). BAS Speech Science Web Services - an Update of Current Developments. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia. ELRA.

Klepsch, M., Schmitz, F., and Seufert, T. (2017). Development and Validation of Two Instruments Measuring Intrinsic, Extraneous, and Germane Cognitive Load. *Frontiers in psychology*, 8:1997.

Kohavi, R. and John, G. H. (1997). Wrappers for Feature Subset Selection. *Artificial Intelligence*, 97(1-2):273–324.

Kramer, A. F. (1991). Physiological Metrics of Mental Workload: A Review of Recent Progress. In *Multiple-task performance*, pp. 279–328. CRC Press.

Krell, M. (2015). Evaluating an instrument to measure mental load and mental effort using Item Response Theory. *Science Education Review Letters*.

Kua, J. M. K., Sethu, V., Le, P. N., and Ambikairajah, E. (2014). The UNSW Submission to INTERSPEECH 2014 ComParE Cognitive Load Challenge. In *Proceedings of INTERSPEECH 2014*, pp. 746–750, Singapore. ISCA.

Kun, A. L., Medenica, Z., Palinko, O., and Heeman, P. A. (2011). Utilizing Pupil Diameter to Estimate Cognitive Load Changes During Human Dialogue: A Preliminary Study. In *Proceedings of the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, Salzburg Austria.

Lass, N. J. (2014). *Speech and Language: Advances in Basic Research and Practice*, volume 5. Academic Press.

Le, P. N. (2012). *The Use of Spectral Information in the Development of Novel Techniques for Speech-Based Cognitive Load Classification*. PhD thesis, University of New South Wales, Sydney, Australia.

Le, P. N., Ambikairajah, E., Choi, E. H., and Epps, J. (2009). A non-uniform subband approach to speech-based cognitive load classification. In *Proceedings of the 7th International Conference on Information, Communications and Signal Processing (ICICS 2009)*, Macau, China. IEEE.

Le, P. N., Ambikairajah, E., Epps, J., Sethu, V., and Choi, E. H. (2011). Investigation of spectral centroid features for cognitive load classification. *Speech Communication*, 53(4):540–551.

Le, P. N., Epps, J., Ambikairajah, E., and Sethu, V. (2010a). Robust Speech-Based Cognitive Load Classification Using a Multi-band Approach. In *Proceedings of the APSIPA Annual Summit and Conference (ASC 2010)*, pp. 400–404, Singapore.

Le, P. N., Epps, J., Choi, E. H., and Ambikairajah, E. (2010b). A Study of Voice Source and Vocal Tract Filter Based Features in Cognitive Load Classification. In *Proceedings of the 20th International Conference on Pattern Recognition (ICPR 2010)*, pp. 4516–4519, Istanbul, Turkey. IEEE.

Lee, C. M., Narayanan, S. S., and Pieraccini, R. (2002). Combining Acoustic and Language Information for Emotion Recognition. In *Proceedings of the Seventh International Conference on Spoken Language Processing (ICSLP 2002)*, pp. 873–876, Denver, CO, USA. ISCA.

Lee, K.-F. (1988). *Automatic Speech Recognition: The Development of the SPHINX System*, volume 62. Springer Science & Business Media.

Lefter, I., Rothkrantz, L., Wiggers, P., and Van Leeuwen, D. (2010). Emotion Recognition from Speech by Combining Databases and Fusion of Classifiers. In *Proceedings of the International Conference on Text, Speech and Dialogue (TSD 2010)*, pp. 353–360, Brno, Czech Republic. Springer.

Leppink, J., Paas, F., Van der Vleuten, C. P., Van Gog, T., and Van Merriënboer, J. J. (2013). Development of an instrument for measuring different types of cognitive load. *Behavior Research Methods*, 45(4):1058–1072.

Lerch, A. (2012). *An Introduction to Audio Content Analysis: Applications in Signal Processing and Music Informatics*. John Wiley & Sons.

Lipp, O. V. and Neumann, D. L. (2004). Attentional blink reflex modulation in a continuous performance task is modality specific. *Psychophysiology*, 41(3):417–425.

Liu, F.-H., Stern, R. M., Acero, A., and Moreno, P. J. (1994). Environment normalization for robust speech recognition using direct cepstral comparison. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1994)*, pp. 61–64, Adelaide, SA, Australia. IEEE.

Liu, H. and Motoda, H. (1998). *Feature Selection for Knowledge Discovery and Data Mining*, volume 454. Springer Science & Business Media.

Liu, H. and Motoda, H. (2007). *Computational Methods of Feature Selection*. CRC Press.

Liu, J., Wong, C. K., and Hui, K. K. (2003). An adaptive user interface based on personalized learning. *IEEE Intelligent Systems*, 18(2):52–57.

Lively, S. E., Pisoni, D. B., Van Summers, W., and Bernacki, R. H. (1993). Effects of cognitive workload on speech production: Acoustic analyses and perceptual consequences. *The Journal of the Acoustical Society of America*, 93(5):2962–2973.

Liwicki, M., Graves, A., Fernàndez, S., Bunke, H., and Schmidhuber, J. (2007). A Novel Approach to On-Line Handwriting Recognition Based on Bidirectional Long Short-Term Memory Networks. In *Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR 2007)*, Curitiba, Paraná, Brazil. IEEE.

Low, L.-S. A., Maddage, N. C., Lech, M., Sheeber, L., and Allen, N. (2009). Content based clinical depression detection in adolescents. In *Proceedings of the 17th European Signal Processing Conference (EUSIPCO-2009)*, pp. 2362–2366, Glasgow, UK. IEEE.

Luengo, I., Navas, E., Hernáez, I., and Sánchez, J. (2005). Automatic Emotion Recognition using Prosodic Parameters. In *Proceedings of INTERSPEECH 2005*, pp. 493–496, Lisbon, Portugal. ISCA.

Majumdar, A. and Ochieng, W. (2002). The factors affecting air traffic controller workload: a multivariate analysis based on simulation modeling of controller workload. *Transportation Research Record: Journal of the Transportation Research Board*, (1788):58–69.

Manolakis, D. G. and Ingle, V. K. (2011). *Applied Digital Signal Processing: Theory and Practice*. Cambridge University Press.

Maragos, P., Kaiser, J. F., and Quatieri, T. F. (1993). Energy separation in signal modulations with application to speech analysis. *IEEE Transactions on Signal Processing*, 41(10):3024–3051.

Marchese, R., Bove, M., and Abbruzzese, G. (2003). Effect of cognitive and motor tasks on postural stability in Parkinson's disease: a posturographic study. *Movement Disorders*, 18(6):652–658.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5(4):115–133.

McKeown, G., Valstar, M., Cowie, R., Pantic, M., and Schroder, M. (2012). The SEMAINE Database: Annotated Multimodal Records of Emotionally Colored Conversations between a Person and a Limited Agent. *IEEE Transactions on Affective Computing*, 3(1):5–17.

Meignier, S. and Merlin, T. (2010). LIUM SpkDiarization: An Open Source Toolkit For Diarization. In *Proceedings of the CMU Sphinx Workshop for Users and Developers (CMU-SPUD 2010)*, Dallas, TX, USA.

Mendoza, E. and Carballo, G. (1998). Acoustic analysis of induced vocal stress by means of cognitive workload tasks. *Journal of Voice*, 12(3):263–273.

Mermelstein, P. (1975). Automatic segmentation of speech into syllabic units. *The Journal of the Acoustical Society of America*, 58(4):880–883.

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97.

Mirghafori, N., Fosler, E., and Morgan, N. (1996). Towards robustness to fast speech in ASR. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1996)*, pp. 335–338, Atlanta, GA, USA. IEEE.

Misra, A. (2012). Speech/Nonspeech Segmentation in Web Videos. In *Proceedings of INTERSPEECH 2012*, pp. 1977–1980, Portland, OR, USA. ISCA.

Miyake, A. and Shah, P. (1999). *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control*. Cambridge University Press.

Möller, J. and Müller-Kalthoff, T. (2000). Lernen mit Hypertext: Effekte von Navigationshilfen und Vorwissen. *Zeitschrift für pädagogische Psychologie*, 14(2/3):116–123.

Montacié, C. and Caraty, M.-J. (2014). High-Level Speech Event Analysis for Cognitive Load Classification . In *Proceedings of INTERSPEECH 2014*, pp. 731–735, Singapore. ISCA.

Montgomery, D. C., Peck, E. A., and Vining, G. G. (2012). *Introduction to Linear Regression Analysis*, volume 821. John Wiley & Sons.

Moray, N. (1979). *Mental workload: Its theory and measurement*, volume 8. Springer Science & Business Media.

Moreno, R. and Park, B. (2010). Historical development and relation to other theories. In *Cognitive Load Theory*, pp. 9–28. Cambridge University Press.

Mulder, L. (1992). Measurement and analysis methods of heart rate and respiration for use in applied environments. *Biological Psychology*, 34(2):205–236.

Müller, C., Großmann-Hutter, B., Jameson, A., Rummer, R., and Wittig, F. (2001). Recognizing Time Pressure and Cognitive Load on the Basis of Speech: An Experimental Study. In *Proceedings of the 8th International Conference on User Modeling (UM 2001)*, pp. 24–33, Sonthofen, Germany. Springer.

Nickel, P. and Nachreiner, F. (2000). Psychometric Properties of the 0.1 HZ Component of HRV as an Indicator of Mental Strain. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, pp. 747–750, Oldenburg, Germany. SAGE Publications.

Nwe, T. L., Foo, S. W., and De Silva, L. C. (2003). Detection of stress and emotion in speech using traditional and FFT based log energy features. In *Proceedings of the Fourth International Conference on Information, Communications and Signal Processing and the Fourth Pacific Rim Conference on Multimedia*, pp. 1619–1623, Singapore. IEEE.

Nwe, T. L., Nguyen, T. H., and Ma, B. (2014). On the Use of Bhattacharyya based GMM Distance and Neural Net Features for Identification of Cognitive Load Levels. In *Proceedings of INTERSPEECH 2014*, pp. 736–740, Singapore. ISCA.

Oppenheim, A. V., Schafer, R. W., and Buck, J. R. (1999). *Discrete-time Signal Processing (2Nd Ed.)*. Prentice Hall.

Orlikoff, R. F. and Baken, R. (1989). The Effect of the Heartbeat on Vocal Fundamental Frequency Perturbation. *Journal of Speech and Hearing Research*, 32(3):576–582.

Oviatt, S. (2006). Human-centered Design Meets Cognitive Load Theory: Designing Interfaces That Help People Think. In *Proceedings of the 14th ACM international conference on Multimedia (ACM MM 2006)*, pp. 871–880, Santa Barbara, CA, USA. ACM.

Paas, F., Renkl, A., and Sweller, J. (2003a). Cognitive Load Theory and Instructional Design: Recent Developments. *Educational Psychologist*, 38(1):1–4.

Paas, F., Renkl, A., and Sweller, J. (2004). Cognitive Load Theory: Instructional Implications of the Interaction between Information Structures and Cognitive Architecture. *Instructional Science*, 32(1):1–8.

Paas, F. and Sweller, J. (2014). Implications of Cognitive Load Theory for Multimedia Learning. In *The Cambridge Handbook of Multimedia Learning*, volume 27, pp. 27–42. Cambridge University Press.

Paas, F., Tuovinen, J. E., Tabbers, H., and Van Gerven, P. W. (2003b). Cognitive Load Measurement as a Means to Advance Cognitive Load Theory. *Educational Psychologist*, 38(1):63–71.

Paas, F., Tuovinen, J. E., Van Merrienboer, J. J., and Aubteen Darabi, A. (2005). A Motivational Perspective on the Relation Between Mental Effort and Performance: Optimizing Learner Involvement in Instruction. *Educational Technology Research and Development*, 53(3):25–34.

Paas, F. and Van Merriënboer, J. J. (1994a). Instructional control of cognitive load in the training of complex cognitive tasks. *Educational Psychology Review*, 6(4):351–371.

Paas, F. and Van Merriënboer, J. J. (1994b). Variability of Worked Examples and Transfer of Geometrical Problem-Solving Skills: A Cognitive-Load Approach. *Journal of Educational Psychology*, 86(1):122–133.

Paas, F., Van Merriënboer, J. J., and Adam, J. J. (1994). Measurement of cognitive load in instructional research. *Perceptual and Motor Skills*, 79(1):419–430.

Park, T. H. (2009). *Introduction to Digital Signal Processing: Computer Musically Speaking*. World Scientific.

Pearson, K. (1895). Note on Regression and Inheritance in the Case of Two Parents. *Proceedings of the Royal Society of London*, 58:240–242.

Pelecanos, J. and Sridharan, S. (2001). Feature Warping for Robust Speaker Verification. In *Proceedings of ODYSSEY-2001*, pp. 213–218, Brisbane, Australia. ISCA.

Peterson, L. and Peterson, M. J. (1959). Short-Term Retention of Individual Verbal Items. *Journal of Experimental Psychology*, 58(3):193–198.

Pfister, B. and Kaufmann, T. (2008). *Sprachverarbeitung: Grundlagen und Methoden der Sprachsynthese und Spracherkennung*. Springer.

Piaget, J. (1923). *The language and thought of the child*. New York: Harcourt.

Plass, J. L., Moreno, R., and Brünken, R. (2010). *Cognitive Load Theory*. Cambridge University Press.

Pouw, W. T., Eielts, C., Gog, T., Zwaan, R. A., and Paas, F. (2016). Does (Non-) Meaningful Sensori-Motor Engagement Promote Learning With Animated Physical Systems? *Mind, Brain, and Education*, 10(2):91–104.

Priddy, K. L. and Keller, P. E. (2005). *Artificial Neural Networks: An Introduction*, volume 68. SPIE press.

Pudil, P., Novovičová, J., and Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11):1119–1125.

Quatieri, T. F., Williamson, J. R., Smalt, C. J., Patel, T., Perricone, J., Mehta, D. D., Helfer, B. S., Ciccarelli, G., Ricke, D., Malyska, N., Palmer, J., Heaton, K., Eddy, M., and Moran, J. (2015). Vocal biomarkers to discriminate cognitive load in a working memory task. In *Proceedings of INTERSPEECH 2015*, pp. 2684–2688, Dresden, Germany. ISCA.

R Core Team (2015). R: A Language and Environment for Statistical Computing. Version 3.2.3. R Foundation for Statistical Computing. Retrieved from http://www.R-project.org/. Accessed 26 February 2016.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Rabiner, L. and Juang, B.-H. (1993). *Fundamentals of Speech Recognition*. Prentice Hall.

Rabiner, L. and Sambur, M. (1977). Voiced-unvoiced-silence detection using the Itakura LPC distance measure. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1977)*, pp. 323–326, Hartford, CT, USA. IEEE.

Rabiner, L. and Schafer, R. W. (1978). *Digital Processing of Speech Signals*. Prentice Hall.

Radeck-Arneth, S., Milde, B., Lange, A., Gouvêa, E., Radomski, S., Mühlhäuser, M., and Biemann, C. (2015). Open Source German Distant Speech Recognition: Corpus and Acoustic Model. In *Proceedings of the International Conference on Text, Speech and Dialogue (TSD 2015)*, pp. 480–488, Pilsen, Czech Republic. Springer.

Reichel, U. D. (2012). PermA and Balloon: Tools for string alignment and text processing. In *Proceedings of INTERSPEECH 2012*, Portland, OR, USA. ISCA.

Reyes, M. L. and Lee, J. D. (2008). Effects of cognitive load presence and duration on driver eye movements and event detection performance. *Transportation Research Part F: Traffic Psychology and Behaviour*, 11(6):391–402.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

Rothkrantz, L. J., Wiggers, P., Van Wees, J.-W. A., and Van Vark, R. J. (2004). Voice stress analysis. In *Proceedings of the International Conference on Text, Speech and Dialogue (TSD 2004)*, pp. 449–456, Brno, Czech Republic. Springer.

Ruiz, N., Feng, Q. Q., Taib, R., Handke, T., and Chen, F. (2010). Cognitive Skills Learning: Pen Input Patterns in Computer-Based Athlete Training. In *Proceedings of the International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction (ICMI-MLMI 2010)*, Beijing, China. ACM.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning Internal Representations by Error Propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1*, pp. 318–362. MIT Press.

Ruzanski, E., Hansen, J. H., Meyerhoff, J., Saviolakis, G., and Koenig, M. (2005). Effects of Phoneme Characteristics on TEO Feature-based Automatic Stress Detection in Speech. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2005)*, pp. 357–360, Philadelphia, PA, USA. IEEE.

Ryu, K. and Myung, R. (2005). Evaluation of mental workload with a combined measure based on physiological indices during a dual task of tracking and mental arithmetic. *International Journal of Industrial Ergonomics*, 35(11):991–1009.

Salahuddin, L., Cho, J., Jeong, M. G., and Kim, D. (2007). Ultra Short Term Analysis of Heart Rate Variability for Monitoring Mental Stress in Mobile Settings. In *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 2007)*, pp. 4656–4659. IEEE.

Salhi, L. and Cherif, A. (2013). Robustness of auditory Teager Energy Cepstrum Coefficients for classification of pathological and normal voices in noisy environments. *The Scientific World Journal*, 2013.

Samlan, R. A., Story, B. H., and Bunton, K. (2013). Relation of Perceived Breathiness to Laryngeal Kinematics and Acoustic Measures Based on Computational Modeling. *Journal of Speech, Language, and Hearing Research*, 56(4):1209–1223.

Scherer, K. R., Grandjean, D., Johnstone, T., Klasmeyer, G., and Bänziger, T. (2002). Acoustic correlates of task load and stress. In *Proceedings of INTERSPEECH 2002*, pp. 2017–2020, Denver, CO, USA. ISCA.

Schilperoord, J. (2002). On the Cognitive Status of Pauses in Discourse Production. In *Contemporary Tools and Techniques for Studying Writing*, pp. 61–87. Springer.

Schnotz, W. and Kürschner, C. (2007). A Reconsideration of Cognitive Load Theory. *Educational Psychology Review*, 19(4):469–508.

Schukat-Talamazzini, E. G. (1995). *Automatische Spracherkennung: Grundlagen, statistische Modelle und effiziente Algorithmen*. Vieweg Verlag.

Schuller, B. (2013). *Intelligent Audio Analysis*. Springer.

Schuller, B., Batliner, A., Seppi, D., Steidl, S., Vogt, T., Wagner, J., Devillers, L., Vidrascu, L., Amir, N., Kessous, L., and Aharonson, V. (2007). The Relevance of Feature Type for the Automatic Classification of Emotional User States: Low Level Descriptors and Functionals. In *Proceedings of INTERSPEECH 2007*, pp. 2253–2256, Antwerp, Belgium. ISCA.

Schuller, B., Batliner, A., Steidl, S., and Seppi, D. (2011a). Recognising Realistic Emotions and Affect in Speech: State of the Art and Lessons Learnt from the First Challenge. *Speech Communication*, 53(9):1062–1087.

Schuller, B. and Rigoll, G. (2006). Timing Levels in Segment-Based Speech Emotion Recognition. In *Proceedings of INTERSPEECH 2006 – ICSLP*, pp. 1818–1821, Pittsburgh, PA, USA.

Schuller, B., Rigoll, G., and Lang, M. (2004). Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine-belief network architecture. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2004)*, pp. 577–580, Montreal, Canada. IEEE.

Schuller, B., Steidl, S., Batliner, A., Epps, J., Eyben, F., Ringeval, F., Marchi, E., and Zhang, Y. (2014). The INTERSPEECH 2014 Computational Paralinguistics Challenge: Cognitive & Physical Load. In *Proceedings of INTERSPEECH 2014*, pp. 427–431, Singapore. ISCA.

Schuller, B., Vlasenko, B., Eyben, F., Wollmer, M., Stuhlsatz, A., Wendemuth, A., and Rigoll, G. (2010). Cross-Corpus Acoustic Emotion Recognition: Variances and Strategies. *IEEE Transactions on Affective Computing*, 1(2):119–131.

Schuller, B., Zhang, Z., Weninger, F., and Rigoll, G. (2011b). Selecting Training Data for Cross-Corpus Speech Emotion Recognition: Prototypicality vs. Generalization. In *Proceedings of Afeka-AVIOS Speech Processing 2011*, Tel Aviv, Israel.

Schuller, B., Zhang, Z., Weninger, F., and Rigoll, G. (2011c). Using Multiple Databases for Training in Emotion Recognition: To Unite or to Vote? In *Proceedings of INTERSPEECH 2011*, pp. 1553–1556, Florence, Italy. ISCA.

Schultz, R., Peter, C., Blech, M., Voskamp, J., and Urban, B. (2007). Towards Detecting Cognitive Load and Emotions in Usability Studies Using the RealEYES Framework. In *Proceedings of the International Conference on Usability and Internationalization*, pp. 412–421, Beijing, China. Springer.

Sethu, V., Ambikairajah, E., and Epps, J. (2007). Speaker Normalisation for Speech-Based Emotion Detection. In *Proceedings of the 15th International Conference on Digital Signal Processing*, pp. 611–614, Cardiff, UK. IEEE.

Sexton, J. B. and Helmreich, R. L. (2000). Analyzing Cockpit Communications: The Links Between Language, Performance, Error, and Workload. *Journal of Human Performance in Extreme Environments*, 5(1):63–68.

Shao, L., Shan, C., Luo, J., and Etoh, M. (2010). *Multimedia Interaction and Intelligent User Interfaces: Principles, Methods and Applications*. Springer Science & Business Media.

Shevade, S. K., Keerthi, S. S., Bhattacharyya, C., and Murthy, K. R. K. (2000). Improvements to the SMO algorithm for SVM regression. *IEEE Transactions on Neural Networks*, 11(5):1188–1193.

Shi, Y., Choi, E., Taib, R., and Chen, F. (2009). Designing Cognition-Adaptive Human-Computer Interface for Mission-Critical Systems. In *Information Systems Development*, pp. 111–119. Springer.

Shi, Y., Ruiz, N., Taib, R., Choi, E., and Chen, F. (2007). Galvanic Skin Response (GSR) As an Index of Cognitive Load. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems*, pp. 2651–2656, New York, NY, USA. ACM.

Shiffrin, R. M. and Atkinson, R. C. (1969). Storage and retrieval processes in long-term memory. *Psychological Review*, 76(2):179–193.

Siegle, G. J., Ichikawa, N., and Steinhauer, S. (2008). Blink before and after you think: blinks occur prior to and following cognitive load indexed by pupillary responses. *Psychophysiology*, 45(5):679–687.

Smith, S. (2013). *Digital Signal Processing. A Practical Guide for Engineers and Scientists*. Newnes.

Sotiropoulos, D. N. and Tsihrintzis, G. A. (2017). *Machine Learning Paradigms - Artificial Immune Systems and their Applications in Software Personalization*, volume 118. Springer.

Steeneken, H. J. and Hansen, J. H. (1999). Speech under stress conditions: overview of the effect on speech production and on system performance. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1999)*, pp. 2079–2082, Phoenix, AZ, USA. IEEE.

Strand, O. M. and Egeberg, A. (2004). Cepstral mean and variance normalization in the model domain. In *Proceedings of COST278 and ISCA Tutorial and Research Workshop (ITRW) on Robustness Issues in Conversational Interaction*, Norwich, UK. ISCA.

Stroop, J. R. (1935). Studies of interference in serial verbal reactions. *Journal of Experimental Psychology*, 18(6):643–662.

Svangren, M. K., Skov, M. B., and Kjeldskov, J. (2017). The Connected Car: An Empirical Study of Electric Cars As Mobile Digital Devices. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI 2017)*, Vienna, Austria. ACM.

Svensson, E. A. and Wilson, G. F. (2002). Psychological and Psychophysiological Models of Pilot Performance for Systems Development and Mission Evaluation. *The International Journal of Aviation Psychology*, 12(1):95–110.

Sweller, J. (2010a). Element interactivity and intrinsic, extraneous, and germane cognitive load. *Educational psychology review*, 22(2):123–138.

Sweller, J. (2010b). Recent theoretical advances. In *Cognitive Load Theory*, pp. 29–47. Cambridge University Press.

Sweller, J., Ayres, P., and Kalyuga, S. (2011). *Cognitive Load Theory*, volume 1. Springer.

Sweller, J., Van Merrienboer, J. J., and Paas, F. (1998). Cognitive Architecture and Instructional Design. *Educational Psychology Review*, 10(3):251–296.

Tabbers, H. K., Martens, R. L., and Merriënboer, J. J. (2004). Multimedia instructions and cognitive load theory: Effects of modality and cueing. *British Journal of Educational Psychology*, 74(1):71–81.

Teager, H. and Teager, S. (1990). Evidence for Nonlinear Sound Production Mechanisms in the Vocal Tract. In *Speech Production and Speech Modelling*, pp. 241–261. Springer.

Tolkmitt, F. J. and Scherer, K. R. (1986). Effect of Experimentally Induced Stress on Vocal Parameters. *Journal of Experimental Psychology: Human Perception and Performance*, 12(3):302–313.

Tracy, J. P. and Albers, M. J. (2006). Measuring cognitive load to test the usability of web sites. *Usability and Information Design*, pp. 256–260.

Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., and Zafeiriou, S. (2016). Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016)*, pp. 5200–5204, Shanghai, China. IEEE.

Tucker, R. (1992). Voice activity detection using a periodicity measure. *IEE Proceedings I - Communications, Speech and Vision*, 139(4):377–380.

Tzanetakis, G. and Cook, P. (2002). Musical Genre Classification of Audio Signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302.

Van Gerven, P. W., Paas, F., Van Merriënboer, J. J., and Schmidt, H. G. (2004). Memory load and the cognitive pupillary response in aging. *Psychophysiology*, 41(2):167–174.

Van Gerven, P. W., Paas, F., Van Merriënboer, J. J., and Schmidt, H. G. (2006). Modality and variability as factors in training the elderly. *Applied Cognitive Psychology*, 20(3):311–320.

Van Gog, T. and Paas, F. (2012). Cognitive Load Measurement. In *Encyclopedia of the Sciences of Learning*, pp. 599–601. Springer.

Van Merriënboer, J. J. and Sweller, J. (2010). Cognitive load theory in health professional education: design principles and strategies. *Medical Education*, 44(1):85–93.

Van Segbroeck, M., Travadi, R., Vaz, C., Kim, J., Black, M. P., Potamianos, A., and Narayanan, S. S. (2014). Classification of Cognitive Load from Speech using an i-vector Framework. In *Proceedings of INTERSPEECH 2014*, pp. 751–755, Singapore. ISCA.

Vanitha, L., Suresh, G. R., Chandrasekar, M., and Punita, P. (2017). Development of four stress levels in group stroop colour word test using HRV analysis. *Biomedical Research*, 28(1):98–105.

Verhasselt, J. P. and Martens, J.-P. (1996). A fast and reliable rate of speech detector. In *Proceeding of the Fourth International Conference on Spoken Language Processing (ICSLP 1996)*, pp. 2258–2261, Philadelphia, PA, USA. IEEE.

Viikki, O. and Laurila, K. (1998). Cepstral Domain Segmental Feature Vector Normalization for Noise Robust Speech Recognition. *Speech Communication*, 25(1):133–147.

Vinayagamoorthy, V., Ramdhany, R., and Hammond, M. (2016). Enabling Frame-Accurate Synchronised Companion Screen Experiences. In *Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video (TVX 2016)*, pp. 83–92, Chicago, IL, USA. ACM.

Walker, W., Lamere, P., Kwok, P., Raj, B., Singh, R., Gouvea, E., Wolf, P., and Woelfel, J. (2004). *Sphinx-4: A Flexible Open Source Framework for Speech Recognition*. Sun Microsystems, Inc.

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

Wells, J. (1997). SAMPA computer readable phonetic alphabet. In *Handbook of Standards and Resources for Spoken Language Systems*, volume 4, pp. 684–732. Walter de Gruyter.

Wells, J. (2000). SAMPA (Speech Assessment Methods Phonetic Alphabet), American English. http://www.phon.ucl.ac.uk/home/sampa/american.htm. Accessed 7 February 2017.

Weninger, F., Eyben, F., Schuller, B., Mortillaro, M., and Scherer, K. R. (2013). On the Acoustics of Emotion in Audio: What Speech, Music, and Sound have in Common. *Frontiers in Psychology*, 4:292–304.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560.

Westbrook, A. and Braver, T. S. (2015). Cognitive effort: A neuroeconomic approach. *Cognitive, Affective, & Behavioral Neuroscience*, 15(2):395–415.

Wickens, C. D. (2008). Multiple resources and mental workload. *Human Factors*, 50(3):449–455.

Wierwille, W. W. and Eggemeier, F. T. (1993). Recommendations for mental workload measurement in a test and evaluation environment. *Human Factors*, 35(2):263–281.

Wilpon, J., Rabiner, L., and Martin, T. (1984). An Improved Word-Detection Algorithm for Telephone-Quality Speech Incorporating Both Syntactic and Semantic Constraints. *Bell Labs Technical Journal*, 63(3):479–498.

Wilson, G. F. and Russell, C. A. (2003). Real-Time Assessment of Mental Workload Using Psychophysiological Measures and Artificial Neural Networks. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 45(4):635–644.

Wirzberger, M., Herms, R., Bijarsari, S. E., Rey, G. D., and Eibl, M. (2017). Influences of cognitive load on learning performance, speech and physiological parameters in a dual-task setting. In *Abstracts of the 20th Conference of the European Society for Cognitive Psychology*, p. 161, Potsdam, Germany.

Wirzberger, M., Herms, R., Bijarsari, S. E., Rey, G. D., and Eibl, M. (2018). Cognitive load influences performance, speech and physiological parameters in a multimodal dual-task setting. In *Abstracts of the 60th Conference of Experimental Psychologists*, p. 296, Marburg, Germany.

Witten, I. H., Frank, E., Hall, M. A., and Pal, C. J. (2016). The WEKA Workbench. In *Data Mining: Practical Machine Learning Tools and Techniques*. Fourth Edition. Morgan Kaufmann.

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

Wöllmer, M., Metallinou, A., Eyben, F., Schuller, B., and Narayanan, S. (2010). Context-Sensitive Multimodal Emotion Recognition from Speech and Facial Expression Using Bidirectional LSTM Modeling. In *Proceedings of INTERSPEECH 2010*, pp. 2362–2365, Makuhari, Japan. ISCA.

Woo, K.-H., Yang, T.-Y., Park, K.-J., and Lee, C. (2000). Robust voice activity detection algorithm for estimating noise spectrum. *Electronics Letters*, 36(2):180–181.

Xie, B. and Salvendy, G. (2000). Prediction of Mental Workload in Single and Multiple Tasks Environments. *International Journal of Cognitive Ergonomics*, 4(3):213–242.

Yap, T. F. (2012). *Speech Production Under Cognitive Load: Effects and Classification*. PhD thesis, University of New South Wales, Sydney, Australia.

Yap, T. F., Ambikairajah, E., Choi, E., and Chen, F. (2009). Phase based features for cognitive load measurement system. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009)*, pp. 4825–4828, Taipei, Taiwan. IEEE.

Yap, T. F., Ambikairajah, E., Epps, J., and Choi, E. H. (2010a). Cognitive load classification using formant features. In *Proceedings of the 10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010)*, pp. 221–224, Kuala Lumpur, Malaysia. IEEE.

Yap, T. F., Epps, J., Ambikairajah, E., and Choi, E. H. (2010b). An Investigation of Formant Frequencies for Cognitive Load Classification. In *Proceedings of INTERSPEECH 2010*, pp. 2022–2025, Makuhari, Japan. ISCA.

Yap, T. F., Epps, J., Ambikairajah, E., and Choi, E. H. (2011a). Formant Frequencies under Cognitive Load: Effects and Classification. *EURASIP journal on advances in signal processing*, 2011(1):1–11.

Yap, T. F., Epps, J., Ambikairajah, E., and Choi, E. H. (2011b). Voice source features for cognitive load classification. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2011)*, pp. 5700–5703, Prague, Czech Republic. IEEE.

Yap, T. F., Epps, J., Ambikairajah, E., and Choi, E. H. (2015). Voice source under cognitive load: Effects and classification. *Speech Communication*, 72:74–95.

Yin, B., Chen, F., Ruiz, N., and Ambikairajah, E. (2008). Speech-based cognitive load monitoring system. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2008)*, pp. 2041–2044, Las Vegas, NV, USA. IEEE.

Yin, B., Ruiz, N., Chen, F., and Khawaja, M. A. (2007). Automatic Cognitive Load Detection from Speech Features. In *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces*, pp. 249–255, Adelaide, Australia. ACM.

Young, J. Q., Van Merrienboer, J., Durning, S., and Ten Cate, O. (2014). Cognitive Load Theory: Implications for medical education: AMEE Guide No. 86. *Medical Teacher*, 36(5):371–384.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P. (2006). *The HTK Book*. Cambridge University.

Zhang, T., Hasegawa-Johnson, M., and Levinson, S. E. (2006). Cognitive State Classification in a Spoken Tutorial Dialogue System. *Speech Communication*, 48(6):616–632.

Zhang, X.-L. and Wu, J. (2013). Deep Belief Networks Based Voice Activity Detection. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(4):697–710.

Zhou, G., Hansen, J. H., and Kaiser, J. F. (1998a). Classification of Speech under Stress Based on Features Derived from the Nonlinear Teager Energy Operator. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 1998)*, pp. 549–552, Seattle, WA, USA. IEEE.

Zhou, G., Hansen, J. H., and Kaiser, J. F. (1998b). Linear and Nonlinear Speech Feature Analysis for Stress Classification. In *Proceedings of the Fifth International Conference on Spoken Language Processing (ICSLP 1998)*, Sydney, Australia. ISCA.

Zhou, G., Hansen, J. H., and Kaiser, J. F. (2001). Nonlinear Feature Based Classification of Speech Under Stress. *IEEE Transactions on Speech and Audio Processing*, 9(3):201–216.

Zlotnik, A., Montero, J. M., San-Segundo, R., and Gallardo-Antolín, A. (2015). Random forest-based prediction of Parkinson's disease progression using acoustic, ASR and intelligibility features. In *Proceedings of INTERSPEECH 2015*, pp. 503–507, Dresden, Germany. ISCA.

Zolnay, A., Schluter, R., and Ney, H. (2005). Acoustic feature combination for robust speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 2005)*, pp. 457–460, Philadelphia, PA, USA. IEEE.

Dissertationen der Medieninformatik

(1) Kürsten, Jens (2012)
A Generic Approach to Component-Level Evaluation in Information Retrieval
ISBN 978-3-941003-68-2
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-96344

(2) Berger, Arne (2014)
Prototypen im Interaktionsdesign: Klassifizierung der Dimensionen von Entwurfsartefakten zur Optimierung der Kooperation von Design und Informatik
ISBN 978-3-944640-00-6
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-126344

(3) Ritter, Marc (2014)
Optimierung von Algorithmen zur Videoanalyse – Ein Analyseframework für die Anforderungen lokaler Fernsehsender
ISBN 978-3-944640-09-9
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-133517

(4) Kurze, Albrecht (2016)
Modellierung des QoS-QoE-Zusammenhangs für mobile Dienste und empirische Bestimmung in einem Netzemulations-Testbed
ISBN 978-3-944640-60-0
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-195066

(5) Wilhelm-Stein, Thomas (2016)
Information Retrieval in der Lehre – Unterstützung des Erwerbs von Praxiswissen zu Information Retrieval Komponenten mittels realer Experimente und Spielemechaniken
ISBN 978-3-944640-82-2
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-199778

(6) Schneider, Anke (2016)
Farbeinflussfaktoren zur emotionalen Bildwirkung und ihre Bedeutung für das Retrieval von Tourismusbildern
ISBN 978-3-96100-002-9
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-209553

(7) Rickert, Markus (2017)
Inhaltsbasierte Analyse und Segmentierung narrativer, audiovisueller Medien
ISBN 978-3-96100-029-6
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-226724

(8) Müller, Stefanie (2018)
Systematisierung und Identifizierung von Störquellen und Störerscheinungen in zeithistorischen Videodokumenten am Beispiel digitalisierter Videobestände sächsischer Lokalfernsehsender
ISBN 978-3-96100-052-4
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa2-214115

(9) Herms, Robert (2019)
Effective Speech Features for Cognitive Load Assessment: Classification and Regression
ISBN 978-3-96100-087-6
Volltext: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa2-333464