Institute for Natural Language Processing
University of Stuttgart
Pfaffenwaldring 5B
D-70569 Stuttgart

Masterarbeit

Exploring Simplified Subtitles to Support Spoken Language Understanding

Katrin Angerbauer

Course of Study:	Softwaretechnik
Examiner:	Prof. Dr. Ngoc Thang Vu, Dr. Antje Schweitzer
Supervisor:	Dr. Heike Adel, Prof. Dr. Ngoc Thang Vu
Commenced:	18.06.2018
Completed:	18.12.2018

Acknowledgements

First of all, I would like to thank my supervisors Heike and Thang, who were there for me and my questions throughout the thesis. Thanks for constantly providing me with good advice and helping me out when I got stuck. Though a Master thesis surely requires some work, I really had fun working with you.

Thanks also to Antje for discussing the user study with me in a series of mails, that really helped me. I am also grateful for all the input and support I got from all the other people at the IMS. Thanks to the participants for taking part in my user study. In that context, let me mention quick "thank you" to the "Kaffeekasse" for providing me with change to pay my participants.

Last but not least, a huge thank you to my family and friends, for always being there for me. Thanks for all your advice, proof reading, the cake and all the other treats during my writing time. Finally, a big thanks to my boyfriend Michael, for always having my back and putting a smile on my face even when things got stressful.

Abstract

Understanding spoken language is a crucial skill we need throughout our lives. Yet, it can be difficult for various reasons, especially for those who are hard-of-hearing or just learning to speak a language. Captions or subtitles are a common means to make spoken information accessible. Verbatim transcriptions of talks or lectures are often cumbersome to read, as we generally speak faster than we read. Thus, subtitles are often edited to improve their readability, either manually or automatically.

This thesis explores the automatic summarization of sentences and employs the method of sentence compression by deletion with recurrent neural networks. We tackle the task of sentence compression from different directions. On one hand, we look at a technical solution for the problem. On the other hand, we look at the human-centred perspective by investigating the effect of compressed subtitles on comprehension and cognitive load in a user study. Thus, the contribution is twofold: We present a neural network model for sentence compression and the results of a user study evaluating the concept of simplified subtitles.

Regarding the technical aspect 60 different configurations of the model were tested. The best-scoring models achieved results comparable to state of the art approaches. We use a Sequence to Sequence architecture together with a compression ratio parameter to control the resulting compression ratio. Thereby, a compression ratio accuracy of 42.1 % was received for the best-scoring model configuration, which can be used as baseline for future experiments in that direction. Results from the 30 participants of the user study show that shortened subtitles could be enough to foster comprehension, but result in higher cognitive load. Based on that feedback we gathered design suggestions to improve future implementations in respect to their usability. Overall, this thesis provides insights on the technological side as well as from the end-user perspective to contribute to an easier access to spoken language.

Kurzfassung

Die Fähigkeit gesprochene Sprache zu verstehen, ist ein essentieller Teil unseres Lebens. Das Verständnis kann jedoch aus einer Vielzahl von Gründen erschwert werden, insbesondere wenn man anfängt eine Sprache zu lernen oder das Hörvermögen beeinträchtigt ist. Untertitel erleichtern und ermöglichen das Verständnis von gesprochener Sprache. Wortwörtliche Beschreibungen des Gesagten sind oftmals anstrengend zu lesen, da man weitaus schneller sprechen als lesen kann. Um Untertitel besser lesbar zu machen, werden sie daher manuell oder maschinell bearbeitet.

Diese Arbeit untersucht das automatische Zusammenfassen von Sätzen mithilfe der Satzkompression durch rekurrente neuronale Netzen. Die Problemstellung wird von zwei Gesichtspunkten aus betrachtet. Es wird eine technische Lösung für Satzkompression vorgestellt, aber auch eine nutzerorientierte Perspektive eingenommen. Hierzu wurde eine Nutzerstudie durchgeführt, welche die Effekte von verkürzten Untertiteln auf Verständnis und kognitive Belastung untersucht.

Für die technische Lösung des Problems wurden 60 verschiedene Modellkonfigurationen evaluiert. Die erzielten Resultate sind vergleichbar mit denen verwandter Arbeiten. Dabei wurde der Einfluss der sogenannten Kompressionsrate untersucht. Dazu wurde eine Sequence to Sequence Architektur implementiert, welche die Kompressionsrate benutzt, um die resultierende Rate des verkürzten Satzes zu kontrollieren. Im Bestfall wurde die Kompressionsrate in 42.1 % der Fälle eingehalten.

Die Ergebnisse der Nutzerstudie zeigen, dass verkürzte Untertitel für das Verständnis ausreichend sind, aber auch in mehr kognitiver Belastung resultieren. Auf Grundlage dieses Feedbacks präsentiert diese Arbeit Designvorschläge, um die Benutzbarkeit von verkürzten Untertiteln angenehmer zu gestalten. Mit den Resultaten von technischer und nutzerorientierter Seite leistet diese Arbeit einen Betrag zur Erforschung von Methoden zur Verständniserleichterung von gesprochener Sprache.

Contents

1 Introduction **17**

1.1 Goals and Contributions of this Thesis 18

1.2 Structure of this Thesis 19

2 Background **21**

2.1 Neural Networks and Deep Learning 21

2.1.1 Basic Concepts of Artificial Neural Networks 21

2.1.2 Long-Short Term Memory Networks and Seq2Seq Architectures 24

2.2 Automatic Text Summarization and Simplification 26

2.2.1 Summarization of Text and Sentences 28

2.2.2 Sentence Compression 29

2.3 Subtitles for Spoken Language Understanding 31

2.3.1 Cognitive Foundations for Subtitle Processing 31

2.3.2 Subtitle Generation and Applications 32

2.4 Evaluation Measures 34

3 Related Work **37**

3.1 Text Summarization and Simplification with Neural Networks 37

3.2 Beyond the Verbatim Subtitle Design: Approaches and Effects 41

3.2.1 Subtitle Design Approaches 41

3.2.2 Studies on the Effect of Partial Captions 43

4 Our Neural Network Model **45**

4.1 Architecture 45

4.1.1 Simple-LSTM 45

4.1.2 Encoder-Decoder Architecture 51

4.2 Implementation Frameworks and Tools 52

4.3 Experiments 53

4.3.1 Datasets and Data Preparation 53

4.3.2 Model Configurations 57

4.3.3 Training Process and Parameters 58

4.3.4 Evaluation Results and Discussion 59

5 User Study **71**

5.1 Methodology 71

5.2 Apparatus 73

5.3 Procedure 75

5.4 Results 78

5.4.1 Comprehension 78

5.4.2 Cognitive Load 78

5.4.3 Subjective Scores 81

5.4.4 Concluding questions 86

5.5 Limitations 87

6 Discussion **89**

6.1 Findings of the User Study 89

6.2 Lessons learned 92

7 Conclusion and Future Work **95**

A User Study Resources **97**

A.1 Video Resources 97

A.2 Example Subtitle Files 98

A.3 Example Questionnaire 114

B Further Experiment Results **147**

Bibliography **151**

List of Figures

1 A Multi-Layer Network. 23

2 Multimedia learning as proposed by Mayer and Moreno (2003) in the case of subtitled videos (c.f. Guillory (1998).) 32

3 Visualization of selection performance of a system. Yellow denotes the relevant information, grey donates irrelevant. TP, FP, FN and TN are defined as mentioned above. 35

4 The architecture of the *Simple-LSTM* model without the additional embedding layer for POS. 46

5 The processing pipeline for POS and word inputs (for a sentence with four words), i.e. a four dimensional row vector.The inputs are processed in separate embedding layers and the word embeddings (orange) and the POS embeddings (green) are then concatenated for further processing in the LSTM layer. 48

6 Combination of the hidden states (grey) with one-hot compression ratio vectors for further processing (dark blue). 49

7 The processing of the hidden states (grey) which were combined with the compression ratios in 6 together with the previous labels (red), visualized for the first two inputs. 50

8 The architecture of the Seq2Seq model without the additional embedding layer for POS. 52

9 Training process with preparation steps. 59

10 Analysis of the accuracy of the compressions according to their compression ratio classes in the evaluation data, tested on the model *Seq2Seq-LSTM_previous_compression* (`no_punct, 256`). 65

11 Analysis of the accuracy of the compressions when assigning one specific compression ratio class for all sentences in the evaluation data, tested on the model *Seq2Seq-LSTM_previous_compression* `(no_punct, 256)`. The correctly predicted compression ratios refer to the whole evaluation dataset. . . . 66

12 Screenshot of a subtitled video. 74

13 Overview of cognitive load based on the aggregated data per participant. The values at the denote the mean of the aggregated data. 80

14 Overall scores of s2. 82

15 Results to *"I would like to have subtitles for the following content"*. 86

List of Tables

1 Distribution of the punctuation characters in the datasets. . . 60

2 Performance with and without the last punctuation character. 60

3 Results for all configurations of model *Simple-LSTM_plain* on the evaluation data set. The best scores are highlighted in bold. 61

4 Best configurations in terms of sentence accuracy per model variation on the evaluation data set. The best sentence accuracy is depicted bold. 62

5 Best configurations in terms of compression accuracy per model variation on the eval data set. The best achieved accuracy is bold. 63

6 Models of the `(no_punct, 256)` configuration compared, results on evaluation dataset. Best values denoted in bold. 64

7 Our models compared to state of the art approaches. "Baseline" refers to the respective implementation of Filippova et al. (2015). Best scores are again denoted bold. 70

8 Overview over the mothertongue of the participants. 72

9 Exposure to English Content (Audio and Video) and Subtitle Usage of Participants 73

10 Example of a sentence in the subtitle files from the different conditions. Line breaks were added only in this Table for presentation purposes. 75

11 Cognitive Load Questions inspired by the cognitive load categories of NASA TLX. 77

12 Subjective Feeback Questions. 77

13 The overall descriptive statistics for the cognitive load categories, the highest values denoted bold. 79

14 Descriptive statistics of questions s1 to s3. 83

15 Descriptive statistics of questions s4 to s6, the highest values in bold. 85

1 Introduction

"Words are, in my not so humble opinion, our most inexhaustible source of magic...", - Albus Dumbledore[1]

We speak approximately 16 000 words per day (Mehl et al., 2007), but probably are confronted with a lot more spoken information we in turn have to listen to. In the morning we have a conversation with the local coffee shop owner to pay our morning coffee. Then we go to work where we have to listen to a presentation in meeting and talk to our colleagues. On the commute home we have to pay attention to the spoken announcements on the train platform and before we go to bed we watch the evening news or some films. These are just a few examples from our daily lives.

In short, spoken speech is ubiquitous. The understanding of spoken speech, however, is not. Though it is a crucial skill needed in everyday situations, it can be impeded by numerous factors. Reasons range from loud background noises, which are a nuisance for everyone independently from their hearing or language capabilities, to the case of language learners and hearing impaired (Krejtz et al., 2016; Vanderplank, 1988).

Subtitles are an assistive technology used in those cases to make spoken content more accessible by transferring oral information to the visual channel by transcription(Burnham et al., 2008). However, reading verbatim subtitles can be cumbersome, as we generally speak faster than we read (Williams and Thorne, 2000). Thus, subtitles are often edited to enable more comfortable reading. Manual editing, however, is time-consuming and people need to be trained especially for that task, which can be expensive. Also, human captioners are often not experts in the topics they are captioning and extracting the important information is difficult for them (Wald, 2006). This circumstance make online editing of talks or other live situations really hard for human captioners and as result more difficult to understand for people relying on

[1] JK Rowling, from Harry Potter and the Deathly Hallows

easy to read captions. Therefore, exploring systems which automatically learn what is important and edit the content directly could make access to spoken information easier and as a consequence available to more people. Such systems then could be used as assistive technology during university lectures, talks or meetings for those who would have difficulty understanding what is said otherwise.

To automatically compress sentences of subtitles and thereby simplify them, one can use sentence compression algorithms (Clarke and Lapata, 2006). Automatic sentence compression, like other Natural Language Processing (NLP) tasks improved in performance with the employment of neural networks, which this thesis uses as well. In the following sections, the goals and the remaining structure of the thesis are outlined.

1.1 Goals and Contributions of this Thesis

This thesis approaches the topic of simplifying spoken language understanding from two perspectives: the technical view and the human-centred view. On the one hand, we wanted to explore state of the art methodologies for sentence compression by implementing a neural network model for the task. On the other hand we wanted to go beyond the mere technical evaluation and also test the effects of compressed subtitles in a user study.

Our neural network model tackles the modelling of the compression ratio parameter, which specifies how much of a sentence is kept in its compressed form. We tested the influence of this parameter on the model performance and evaluated our model against state of the art. According to Zanón (2006) subtitles are a *"dynamic and rich source of communicative language use"*, which is why we wanted to apply our model to subtitle data. Hereby, we wanted to investigate the potential for future application scenarios of sentence compression to foster spoken language understanding.

In order to support spoken language understanding a mere technological contribution is not enough. One has to take into account human capabil-

ities as well. For that reason we conducted a user study to find out more about the effects of compressed subtitles on comprehension and cognitive load. Furthermore we wanted to gather feedback about the perceived usefulness of the compressed subtitles. We evaluated our system compressed subtitles against full subtitles and human compressed subtitles to measure effects of the concept itself on the one hand and to compare the effect of system compressed subtitles against human compressed subtitles to get a more in-depth system evaluation.

From the technical point of view, our model could be used as a starting point for further investigations into the compression ratio parameter and the application of sentence compression to spoken language. The user study showed that idea of simplified subtitles has potential, but one has to take care in the implementation to avoid additional cognitive load. In short, the thesis provides insights into the technological and the end user perspective, which contribute to future research to make spoken language more accessible.

1.2 Structure of this Thesis

The remainder of this thesis is structured as follows. **Section 2** describes the relevant concepts needed to understand the content of the thesis. **Section 3** presents related work done in the field of neural summarization and simplification models as well related research on modified subtitles. **Section 4** introduces the implemented neural network model and its technical evaluation results. On the other hand, **Section 5** deals with the human evaluation in form of a user study. The results of the user study are discussed in **Section 6** and consequences for the implemented prototype are drawn. Finally, **Section 7** concludes this thesis and proposes future research directions.

2 Background

Here, the background for understanding the topic of the thesis is given. Underlying concepts and terminology are explained so that the reader is able to follow the later chapters of the thesis without expert knowledge of the topic. However, it should be mentioned, that detailed and in-depth explanations are beyond the scope of the thesis and relevant content is only touched briefly.

Section 2.1 gives an overview of the basic concepts and terminology of neural networks and deep learning, while Section 2.2 presents the basic approaches to automatic text summarization and simplification. In Section 2.3, the foundations of subtitles are explained. In the last section, Section 2.4, the evaluation measures used in this thesis are introduced.

2.1 Neural Networks and Deep Learning

This section gives a brief introduction to the basic concepts (see Section 2.1.1) of deep learning and presents some of the most common neural networks used today in large variety of tasks (Section 2.1.2).

2.1.1 Basic Concepts of Artificial Neural Networks

When performing a large amount of sequential computations like addition or multiplications computers easily outperform humans. Tasks, however, that seem intuitive and simple to us, like recognizing spoken speech or object recognition, are hard to solve for computers. This problem is owing to the fact that our knowledge of the the world is fairly inherent and based on our subjective experiences, which are fuzzy and hard to express in equations and formalisms (Goodfellow et al., 2016; Rashid, 2016).

Deep Learning is an approach to machine learning, which models the world as hierarchy of concepts. This paradigm allows machines to learn from experience and build up complex problems out of simple ones. It utilizes basic

techniques from statistics and applied maths and its fundamental construct, the artificial neural network, is motivated by the biological brain (Goodfellow et al., 2016). We have approximately 100 billion neurons in our brain, that are interconnected and communicate with each other through electrical signals. Analogously, artificial neurons are connected in a artificial neural network and "communicate" via signals i.e. the output of their calculations (Rashid, 2016).

The most basic building block of an neural network is a single neuron. A perceptron (Rosenblatt, 1958) is one type of artificial neuron that takes binary inputs and computes binary outputs by calculating the weighted sum of the input. Thus, a perceptron or in general, any kind of artificial neuron, decides by *"weighing up evidence"* (Nielsen, 2015). So, the output $y$of a perceptron is determined by:

$$y(x) = \begin{cases} 0 & \text{if } \sum_j w_j x_j \ + b \leqq 0 \\ 1 & \text{if } \sum_j w_j x_j \ + b > 0, \end{cases}$$

where w_j is the weight assigned to input x_j, determining how important it is for the output. Further, b is the *bias* term, which defines how easy it is to output 1,i.e. a strong signal. Thus, those are the parameters which specify the behaviour of a neuron and in which the learned information is stored (Kågebäck et al., 2014; Nielsen, 2015). The output function is also called *activation function*. It transforms the input of a neural network to the output signal, thus determining the *firing* behaviour (Nielsen, 2015; Rashid, 2016).

The problem with the above described activation function is that a small variation of bias or weights can cause the output to flip. That is why smoother activation functions are required. A common activation function is the *sigmoid function* σ, where small changes in the weights and biases result in small changes of the output (Nielsen, 2015). So a more general formulation for the output y of neuron can be defined as follows (the weighted sum is

written as the dot product of the weight vectors and input vectors, b is the bias and f denotes the activation function):

$$y = f(w \cdot x + b)$$

Typically, neural networks consist of more than one neuron. Thereby, they are able to solve more complex computations such as hand-writing recognition. The design of a multi-layer network is shown in Figure 1. In the input layer the input is encoded into neurons.The hidden layers are responsible for creating intermediate representations of the input. Finally the output layer computes the final output that is emitted (Goodfellow et al., 2016; Nielsen, 2015).

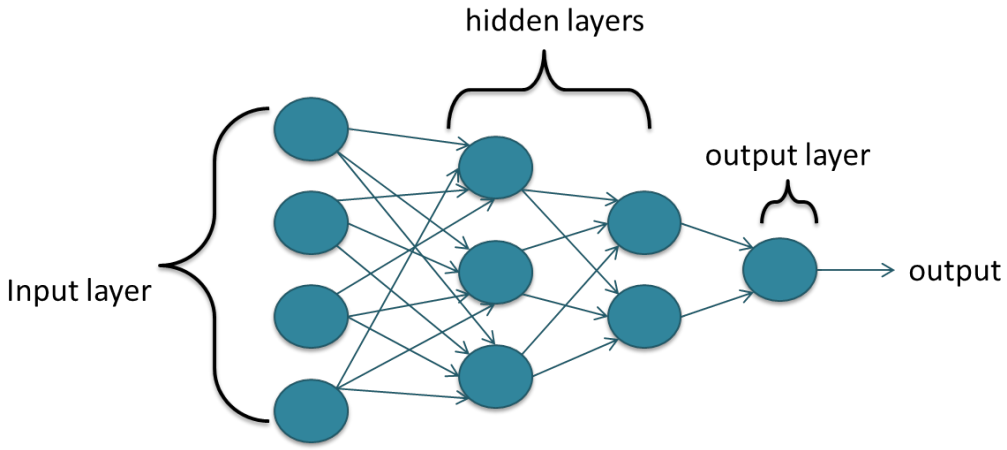

Figure 1: A Multi-Layer Network.

When training a neural network, the goal is to find values for the weights and biases so that the network produces the desired output y for input x. How well a network approximates the desired output can be seen when one calculates the error or loss, i.e. the difference between the output of the network and the desired target output. The target is given by the labelled training data, which makes training a neural network a supervised task.

To rephrase, the goal of training is to minimize the network loss. This loss can be seen as a differentiable function of the output produced by the network and the target output, dependent on the model weights. How the

function looks in detail depends on the network architecture choices. For the minimization of the loss function the gradient $\frac{\partial loss}{\partial parameters}$ is calculated through backpropagation and then minimized with gradient decent (Kågebäck et al., 2014; Nielsen, 2015; Rashid, 2016). For a detailed explanation of those algorithms the reader is asked to consult background literature, as a detailed explanation would be beyond the scope of this thesis.

As a summary, a neural network is a construct of multiple artificial neurons which are interconnected in different layers. It learns from the training data and adapts its parameters during training to achieve better performance (Nielsen, 2015).

2.1.2 Long-Short Term Memory Networks and Seq2Seq Architectures

There exist various types of neural network architectures, the basic one shown in Figure1 in the previous section is called a feed-forward network, where input from the previous layer is used in the subsequent layer. In this kind of network there are no loops, thus only connections to those subsequent layers are allowed (Kågebäck et al., 2014; Nielsen, 2015).

We, however, will only discuss those relevant for this thesis, the Recurrent Neural Net (RNN) and one of their special implementations, the Long-Short Term Memory (LSTM) network. RNNs are networks that have feedback loops and are able to process inputs in form of sequences.Their hidden states store also information on previously seen data.Thus, the computation of the hidden state h_t is not only dependent on the current input x_t but also on the previous hidden state h_{t-1}. Thus we have the following general equation for basic RNN:

$$h_t = f(h_{t-1}, x_t) \text{ usually specified as } h_t = \sigma(Uh_{t-1} + Vx_t),$$

with U and V being weight matrices (Dong, 2018; Nielsen, 2015). There are special types of RNN such as Gated Recurrent Unit RNN or Long-Short-Term Memory Networks (LSTM), able to deal with long term dependencies.

For the sake of brevity and because an LSTM is used in the model below, we will only discuss LSTMs in more detail. The concept of LSTMs was first invented by Hochreiter and Schmidhuber (1997).

It is defined by the following equations (using the notation of Olah (2015) and Chen (2018)):

(1)	$f_t = \sigma(W_f x_t + U_f h_{t-1})$	(forget gate)
(2)	$i_t = \sigma(W_i x_t + U_i h_{t-1})$	(input gate)
(3)	$\tilde{C}_t = tanh(W_C x_t + U_C h_{t-1})$	(candidate vectors)
(4)	$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$	(cell state)
(5)	$o_t = \sigma(W_o x_t + U_o h_{t-1})$	(output gate)
(6)	$h_t = o_t \odot tanh(C_t)$	(hidden state)

x_t denotes the input, the different matrices W_z and U_z are parameters of the model (z being a placeholder for the different indice) and $\odot$ denotes element-wise multiplication.

To explain the mechanisms of the different gates and states mentioned in the equations 1 to 6, a more informal perspective on LSTMs is used (for the mathematical and theoretical background the author refers to Hochreiter and Schmidhuber (1997)).

An LSTM can be seen as a neural network in possession of a "long-term memory" (the cell state C_t) and a "working memory " (the hidden state h_t). At each time step when processing the input x_t, the long term memory and the working memory are updated accordingly.

To update the long-term memory or C_t the network has to decide what information is still relevant from the previous cell state C_{t-1} and what parts of the new information are important. Herefore, the forget gate in equation 1 calculates which information to keep and what information to dispose of. To get the new information of the input the candidate vectors (c.f. equation

3) are computed by taking into account the previous hidden state and the current input. For ranking the candidate values according to their importance they are passed through the input gate. The cell state is then updated by combining the remaining information of the old cell state and the relevant new information, see equation 4 (Chen, 2018; Olah, 2015).

The working memory is updated by considering new information as well integrating knowledge from previously seen data (i.e. the long-term memory C_t). First it has to decide on what information to focus on.This is done at the output gate, defined in equation 5. Then, on basis of the result of the output gate, it has to check whether it has already seen something useful and transfer the relevant information from the cell state to the hidden state, c.f. equation 6 (Chen, 2018; Olah, 2015). The new cell state C_t and hidden state h_t are passed along to the next computation step. The final hidden state is considered the output of the LSTM (Chen, 2018; Olah, 2015).

Several neural networks can be combined into more complex neural network architectures. LSTMs are often used in Sequence to Sequence architectures (Seq2Seq), which are also known as encoder-decoder frameworks. Those are first proposed by Cho et al. (2014) and Sutskever et al. (2014) for the task of sentence translation. In a Seq2Seq architecture, the encoder extracts the information of the input and encodes this information into a sequence of hidden states. This information then is passed on to the decoder, which generates the output sequence (c.f. Cho et al. (2014)). It is to be noted that the decoder processes the output of the encoder token by token.

2.2 Automatic Text Summarization and Simplification

We are currently living in an era, where we are confronted with an huge amount of information on a daily basis. In consequence, condensing important information into a summary or making it more accessible through simplification of its content is becoming more and more important (Dong, 2018). Manual methods however, are not sufficient when dealing with an abundance

of data (Dong, 2018) like we do in the age of the internet, where new information practically hides behind every hyperlink and *"big data"* has become a buzzword. Research on automatic text summarization and simplification therefore has become of more and more importance over the last decades.

Text summarization and text simplification are related and similar, but not equal tasks (Dong, 2018; Shardlow, 2014). Approaches can be supervised (i.e. requiring labelled training data from parallel corpora) or unsupervised (not needed labelled training data) (Clarke and Lapata, 2006). Simplification can be defined either as a paraphrasing problem (Glavaš and Štajner, 2015; Xu et al., 2016) or a monolingual translation task, where one translates from complex to simple content (Nisioi et al., 2017; Specia, 2010; Wubben et al., 2012; Zhu et al., 2010).

Text simplification holds potential to make content more accessible to a broader audience by providing reading assistance (Inui et al., 2003) and on the other hand also help Natural Language Processing tasks to achieve better performance (Chandrasekar et al., 1996), thus it is a task worth looking into for various reasons.

A good simplification should be rewritten in a simpler manner, but remain yet grammatical and preserve the key aspects of a text (Xu et al., 2016). Further, a simplification should be logically entailed from the original sentence and should not convey false information (Guo et al., 2018). To check whether these goals are achieved some kind of evaluation is mandatory (Inui et al., 2003).

Simplification entails more tasks than mere deletion of content. Summarization can be defined as a subtask of simplification. Summarization makes content easier to grasp by distilling it to its mayor information. In the following, we only discuss the task of text summarization as well as the summarization on sentence-level, also known as sentence compression.

2.2.1 Summarization of Text and Sentences

Jones (1999) defines summary generation as an reductive operation which transforms the source text into summary text by reducing and generalizing content. The goal is to produce concise and fluent summary text, that contain the key aspects of the text (Nenkova and McKeown, 2012). Such a summary helps the reader to extract relevant information (Kågebäck et al., 2014).

According to Jones (1999) text summarization follows a three-step pipeline:

1. *Interpretation* of the source to a text representation.
2. *Transformation* of this text representation into a summary representation.
3. *Generation* of the summary text out of the summary representation.

The pipeline of Ren et al. (2017) rather focuses on two tasks, sentence scoring and sentence selection which could be placed in between interpretation and transformation and transformation and generation, respectively.

Jones (1999) further introduces the so-called *context factors* of summaries, which fall into three categories: *input*, *purpose* and *output*. The input factor describes properties of the source that is to be summarized and deals with properties like the size of the input (one vs. multiple documents), the language or the subject type (c.f. also Dong (2018)). The purpose factor, as the name would suggest, is concerned with the reason why the summary is created, in which context it will be used and who the audience is. How the resulting summary looks like, is described by the output factor. According to Dong (2018) the properties *extractive* and *abstractive* are important examples of the output. These output factors are actually the main classification of summarization methods.

Extractive summaries *extract* the relevant information from the source in a top-down manner according to Rush et al. (2015), they talk of "crop

and stitch" mechanism, as the summary is created by first singling out the important aspects and then putting them back together to ideally form a grammatical summary construct.

In contrast, abstractive methods are rather a bottom-up approach as new summary content is created bottom-up by generating new summary phrases based on the main idea of the source (Dong, 2018; Rush et al., 2015). Nallapati et al. (2016) see abstractive summarization as a kind of compressed paraphrasing of the main concepts, while using potentially unseen words. Abstractive summarization seems to be a closer approximation to human summary creation (Knight and Marcu, 2002; See et al., 2017).

2.2.2 Sentence Compression

Sentence compression is the creation of a summary on sentence level. The goal is to create a grammatically correct summary sentence, which is condensed to the main information and ideally unimportant content is deleted (Cohn and Lapata, 2008; Jing and Hongyan, 2000). An example from the Google Sentence Compression Data Set [2]:

> **Sentence:** medical researchers at the university of alberta have discovered the structure of a potential drug target for a rare genetic disease, paving the way for an alternative treatment for the condition.
>
> **Compression:** medical researchers have discovered the structure of a potential drug target for a rare genetic disease

Sentence compression can also be seen as the first step towards sentence simplification (Siddharthan, 2015). It is a form of simplification achieved by deletion of unnecessary content (See et al., 2017).

There exist extractive as well as abstractive approaches. While extractive approaches focus on the deletion of unimportant information in the sentence

[2] https://github.com/google-research-datasets/sentence-compression

(Jing and Hongyan, 2000), abstractive approaches employ other strategies like substitution, reordering or insertion as well to create the summary sentence (Cohn and Lapata, 2008).

Jing and Hongyan (2000) is one of the first to introduce a sentence reduction system, which removes single words or entire grammatical unities from a sentence based on its syntactic parse tree, context information and corpus statistics. The work of Knight and Marcu (2002) and Cohn and Lapata (2008; 2009) provide early methods for abstractive sentence compression. The first is a probabilistic approach on sentence compression, where they employ a noisy-channel-framework, saying that sentence compression is basically the identification of the most essential content, before the other parts of the sentence are been added, i.e. the noise (Knight and Marcu, 2002). The second model of Knight and Marcu (2002) is a tree-based parsing approach as well, based on a shift-crop operation. Cohn and Lapata (2008; 2009) rely on a transducer as well.

Cohn and Lapata (2007) also present a tree-based extractive method for sentence compression, where a parse tree of a sentence is rewritten into the compressed parse tree. The rewrite rules are learned from a parsed corpus. The approach of Filippova and Strube (2008) is also using parse trees. Their method, however, is based on the tree resulting from dependency parsing instead of the syntactical parse tree.

Sentence compression can also be seen as optimization problem, which Clarke and Lapata (2008) aim to solve with an integer linear programming approach. Regardless of all the different models and approaches, the overall goal of sentence compression is to condense a sentence to its most relevant information, while not modifying its meaning.

2.3 Subtitles for Spoken Language Understanding

This section gives an overview of the concept of subtitles.

Some literature differentiates between subtitles being in a different language than the soundtrack and captions being in the same language (Markham, 1999), the latter specifically to assist the hearing impaired. We, however, refer to the terms interchangeably and define subtitles according to Williams and Thorne (2000) as *intralingual* when soundtrack and subtitles are in the same language and *interlingual* when soundtrack and subtitle language differ.

2.3.1 Cognitive Foundations for Subtitle Processing

Reading subtitles is different than reading static texts. The reader is additionally confronted with video and sound, stimuli that potentially compete with one another, because our visual and audio processing capacities are limited. Further, there is no option to read content again to disambiguate the meaning, as it is presented only for a limited amount of time. This can result in high cognitive load(Baddeley, 1992; Guillory, 1998; Koolstra et al., 2002; Krejtz et al., 2016; Moran, 2012).

Mayer and Moreno (2002; 2003) present a model for multimedia learning. This model is based on the following three principles:

- Humans have different channels to process different information modalities, i.e. visual and verbal channels.
- The capacity of these channels is limited.
- Learning requires active processing of the information of these channels.

First, stimuli are perceived through our eyes and ears. Then we decide to pay attention to relevant words and sounds. These words and sounds are then converted into mental models for the respective stimuli in our working memory. Finally, to comprehend the input as a whole construct we merge

the different mental models into one and integrate prior knowledge from our working memory. Cognitive overload occurs if the cognitive demand is too high for the processing capacities available at the different channels (Mayer and Moreno, 2003).

Guillory (1998) mentions a similar model in respect to subtitle processing, where the input of four different stimuli has to be processed on different channels and put into according abstract schemas to lead to comprehension. According to this model, the stimuli are processed in parallel. However, if a processing demand on one channel is too high, the formerly parallel tasks are handled sequentially and information is lost, thus the viewer struggles to comprehend the content of the subtitled video. Figure 2 shows the process of multimedia learning as described by Mayer and Moreno (2003) with the stimuli of subtitled videos mentioned by Guillory (1998).

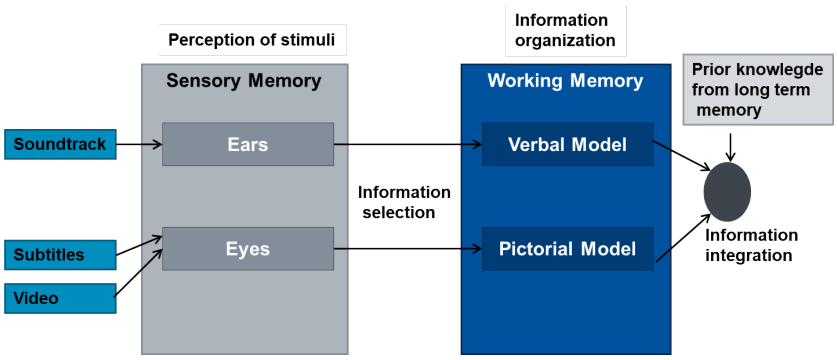

Figure 2: Multimedia learning as proposed by Mayer and Moreno (2003) in the case of subtitled videos (c.f. Guillory (1998).)

The reading skill of the person processing the subtitle also has an influence on the comprehension of the latter (Burnham et al., 2008).

2.3.2 Subtitle Generation and Applications

Subtitles can be seen as an assistive technology, which is based on text-presentation with the aim to improve the accessibility to audio based content

(Burnham et al., 2008). This is especially relevant to the deaf and hard-of-hearing, but also helpful for language learners and people having to understand audio in noisy environments or to understand people with strong accents (Krejtz et al., 2016; Vanderplank, 1988).

For language learners, watching videos with subtitles can have multiple benefits. For one, facilitate the process of following the story of a film. Further, they help to focus the attention. The learners also develop skills for reading rapidly learn new vocabulary and improve their word recognition capabilities (King, 2002; Winke et al., 2010).However, subtitles can tempt the learners to lean on their reading abilities too much, and use them as a support to understand the content, rather then training their listening skills (King, 2002; Winke et al., 2010).

Apart from being inter- or intralingual, subtitles can also be distinguished based on their manner of reflecting the content: Verbatim subtitles transcribe the audio word by word (Guillory, 1998). However, we speak faster than we read, so often edited subtitles are created, where the speech is simplified and compressed up to one third of the content (Ward et al., 2007; Williams and Thorne, 2000).

Williams and Thorne (2000) propose the following guidelines for manual subtitle creation, which could also be seen as design requirements for automatic subtitling systems:

- The subtitles should be easy to read and at the same time transmit the full content.
- The style of the spoken language should be mirrored in the captions.
- The display of the subtitles should be consistent and smooth to avoid confusion.
- The syntax of the subtitles should remain intact.

2.4 Evaluation Measures

This section gives a brief overview of the evaluation measures of this thesis. To evaluate the system we on the one hand calculate different kinds of accuracy and the F1-score. The accuracy (i.e. the number of correctly classified items in relation to the total items) is measured on sentence, token and compression ratio level. Consequently the sentence accuracy (A_s) is calculated by $A_s = \frac{s_c}{S}$, where s_c are the correctly predicted sentences and S the total amount of sentences. A sentences is correctly predicted, if the entire target compression can be reproduced. Token accuracy (A_t) measures how much words (i.e. tokens) are correctly predicted in relation to the total number of words. The so-called compression ratio, a value between one and zero, specifies how much words are kept in the compressed sentence (Cohn and Lapata, 2008), i.e. if a sentence is ten words long and the compression ratio is specified as 0.4 then four words should be in the resulting compressed sentence. The compression ratio accuracy (A_c) is specified by comparing the compression ratio of the resulting compression to a target compression ratio. This target compression ratio is either given by the attributes in the data or specified by the experimenter as desired target for all sentences.

In the field of Information Retrieval (among others) the effectiveness of a system is measured additionally with the measures of *precision* and *recall*(Manning et al., 2009). Precision in our case defines how many words in the compression actually are relevant for the content of the sentence, or in other words, how many words of the resulting compression are in the target compressions as well. The recall measures how much of the relevant items (in our case, words that should be inside the compression) are actually retrieved or selected by the system. In terms of *true positives TP* (selected and relevant), false positives FP (selected but not relevant) and false negatives FN (not selected, but relevant) those are defined as follows using the notation of Manning et al. (2009):

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Figure 3 visualizes the connection between TP, FP, TN and FN. TN are known as the true negatives, the information not relevant and not selected.

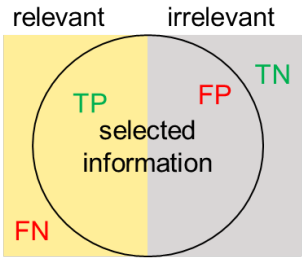

Figure 3: Visualization of selection performance of a system. Yellow denotes the relevant information, grey donates irrelevant. TP, FP, FN and TN are defined as mentioned above.

Between those measures there exists a certain trade-off : Recall increases if you select more, which is usually reciprocal for precision. So, one needs some way to balance those measures. One approach to achieve this, is the F1-score (Manning et al., 2009). The F1-score is the harmonic mean of precision and recall and defined as follows:

$$\mathrm{F1} = \frac{2 * \mathrm{precision} * \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}$$

Usually, one does only calculate the F1-score of the relevant class (in our case the words kept in the compression, i.e. the class KEEP), c.f. Filippova et al. (2015). We, however, report both F1 for KEEP ($f1_K$) and DELETE ($f1_D$). DELETE denoting the words removed from the sentence.

Further, we use measures like mean (M), standard deviation (SD), median and mode for the descriptive analysis of the user study results. The mean is the average value of the data points, the median is the data point in the middle of the distribution and the mode is the most frequent value of the data points. The standard deviation denotes how accurately a mean represents the underlying data points (Field and Hole, 2003).

3 Related Work

In this section related research in the fields of neural networks and subtitles is presented. Section 3.1 deals with related models regarding summarization and sentence compression. We mainly present extractive approaches as they are most relevant for our model as well as abstractive approaches using the same architecture or similar implementation designs.

Section 3.2 on the other hand presents related subtitle design approaches and their effects. Here, we chose a variety of different design approaches to give the reader a broad overview of the design space of subtitles, which as our approach aim to simplify the understanding of spoken speech. Regarding the studies of the effects, we focus on the studies concerning keyword or partial captions, as the effects described there are most interesting in respect to our user study, which also is concerned with partial captions.

3.1 Text Summarization and Simplification with Neural Networks

While Section 2.2 deals with general concepts and early methods, this section presents more recent developments in automatic text summarization and simplification with neural networks, which are used for multiple NLP tasks and produce better results than other approaches without extensive human involvement (Dong, 2018).The summarization pipeline with neural networks is as follows according to Dong (2018):

1. Words are converted into word embeddings by a look-up table (which usually is pre-trained).
2. The encoder model processes word embeddings to create a sentence level representation.
3. Sentence representations are passed to a model responsible for sentence selection.

Extractive approaches are based on the appropriate selection of content and rely on the design decision regarding sentence representations and sentence selection. Abstractive solutions, on the other hand, are centred around the tasks of document representation and word sequence generation (Dong, 2018)

Nallapati et al. (2016) use a feature-rich encoder in their abstractive summarization model, which takes into account part of speech tags (POS tags) and term-frequency inverse-document-frequency (tf-idf) bins as well. Our model is not abstractive. However, we also explore the use of POS tags additionally to the sentence input for our model, similar to Nallapati et al. (2016).

Earlier work on sentence simplification with a sequence to sequence architecture is done by Nisioi et al. (2017). Their architecture can perform extractive and abstractive methods for sentence simplification. Our model, as discussed later, also uses a Seq2Seq architecture, we however put our focus on extractive methods to simplify our sentences by deletion of not relevant content.

The previous models discussed abstractive summarization and simplification, tasks which are closely related. Extractive summarization or compression focuses on deletion operations and could be described as a subtask in the approaches above.

One of the first to develop an extractive approach to summarization with neural networks was Kågebäck et al. (2014). However, their model works on multi-document-level, which is totally in contrast to our model, as our model operates on sentence-level, following the approach of Filippova et al. (2015).

The approach of Filippova et al. (2015) uses multilayer LSTMs to determine the deletion sequence of a sentence. Thus, they treat sentence compression as a sequence labelling task, where a binary decision is made regarding every word in the sentence (KEEP (1) or DELETE (0)). Three different LSTM models are presented by them: A basic one with just the words of

the sentence as input, another with the dependency-parsed input with additional information about the parent word (in respect to the dependency tree), named LSTM+PAR, and their final model with further information about the resulting label of the parent word (named LSTM PAR+PRES). They use a multilayer-LSTM with three stacked LSTM layers, which are preceded by an embedding layer. After the LSTM layers the hidden states are projected into label space with a linear layer and finally a SoftMax Classifier computes the label probabilities. For our basic architecture we do not use a stacked LSTM but rather a bidirectional one, otherwise we follow the implementation of Filippova et al. (2015).

The approaches to sentence compression by Klerke et al. (2016), Tran et al. (2016), Lu et al. (2017) and Lai et al. (2017), like our approach, also use the model from Filippova et al. (2015) as baseline and use their sentence compression dataset for training and test. Their implementation details,contrasts and similarities to our approach are discussed below.

Klerke et al. (2016) use the three-layer LSTM approach of Filippova et al. (2015) and extends it with an multi-task learning mechanism to take into account sentence as well as gaze data, both from different corpora. Multi-task learning describes a parallel training of related tasks in varying minibatches depending on a mixing ratio (Guo et al., 2018). They use a smaller embedding and hidden size than Filippova et al. (2015) and our approach.

Instead of multi-layer LSTMs Tran et al. (2016) uses bidirectional LSTMs to build on top of Filippova et al. (2015). Further they integrate an attention mechnism in the encoder to better filter out relevant content. The concept of attention was also used for abstractive summarization e.g. Rush et al. (2015). We also decided to use bidirectional LSTMs (bi-LSTMs) following the concept of Tran et al. (2016), do not however, implement the concept of attention. Further, our sentence preprocessing differs from their method. They parse the original sentences and tokenize them on their own, while we parse the already tokenized entries of the dependency parse tree.

Similar to our second architecture approach, Lu et al. (2017) and Lai et al. (2017) implement encoder-decoder archictectures to extend the approach of Filippova et al. (2015). They both use two encoders to enhance the semantic modeling results and to somehow implement a neural network representation of the *"human re-reading process"*(Lu et al., 2017). In contrast, we do not include two encoders, but rely on additional information such as POS tags or the compression ratio parameter, as discussed later in section 4.1. Additionally, we also rely on the SoftMax classifier as did Filippova et al. (2015). Lu et al. (2017) and Lai et al. (2017) use different classifiers.

In summary, the neural models discussed here, are either extractive or abstractive, and with simplification or summarization as their main task, where they either operate on sentence or document level.To the best of our knowledge, the implementation and training of the model of Filippova et al. (2015) by Andor et al. (2016) sets the state of the art scores for sentence compression.

In our approach, we compress on sentence level, as subtitles should be read sentence by sentence and with our model we aim to extract the essential information of that sentence. In our opinion, an extractive model makes sense in our context, because the process of reading is "extractive" as well to some extend as we tend to skip some words while reading a sentence and yet can make sense of it (Rayner, 1998). Our architecture is loosely based on the approach of Filippova et al. (2015), but we as Tran et al. (2016) use bi-LSTMs and explore a encoder-decoder structure like Lu et al. (2017) and Lai et al. (2017). We further explore the possibility of rich-feature encoders as mentioned by Nallapati et al. (2016), taking into account POS tags as well. Additionally, we explore the potential of the compression ratio parameter.

3.2 Beyond the Verbatim Subtitle Design: Approaches and Effects

Section 2.3 presented the basic concept of subtitles and their cognitive implications. Yet, there are many more designs of subtitles despite verbatim and edited. This section will describe selected approaches trying to facilitate the understanding of spoken language by edited captions and also shed light on the effect of partial subtitles found in related studies.

3.2.1 Subtitle Design Approaches

The approaches shown in the course of this section can be categorized as follows. There are design approaches experimenting with the subtitle position (citepBrown2015, Kurzhals2017),subtitles attempting to convey additional information (Berke et al., 2017; Piquard-Kipffer et al., 2015; Rashid et al., 2007) and subtitle design approaches trying to condense subtitles to the relevant content (Ferdiansyah and Nakagawa, 2013; Mirzaei et al., 2017; Moran, 2012; Yang et al., 2010). Though the designs are substantially different they all share our goal to facilitate access to spoken speech, and therefore are mentioned to give the reader a brief overview of the great variety of subtitle designs. The works mentioned here do not however, present a complete representation of the design space of subtitles, as this would be beyond the scope of this thesis.

Dynamic Subtitles were investigated by Brown et al. (2015), here captions are placed on different locations near relevant content. Kurzhals et al. (2017) took this idea further and implemented a system in which subtitles are close to the person that speaks and follow her around to minimize the distance between relevant content and the subtitle text.

Rashid et al. (2007)'s approach was dynamic as well to some extent. They developed subtitles that transmit music sound effects as well as prosody through animation and dynamic position. With this mode of captioning,

basic emotions like happiness, sadness, fear, anger and disgust as well as their intensity should be conveyed, which are lost in standard captions. Thus, the approach of Rashid et al. (2007) attempts to help the hearing impaired to grasp the emotional context of scenes in movies. In contrast to this approach, Berke et al. (2017)'s additional information within the captions was not concerned with the content displayed by the captions but rather with caption quality. The proneness to errors by captions done with ASR systems was tackled in this approach, researching how to make potential ASR errors clear in the captions, which otherwise could lead to confusion. They build on top the research of Piquard-Kipffer et al. (2015), who additionally to confidence explored ways to communicate the pronunciation of the words in the subtitles. Both approaches examined whether to highlight the confident parts of the subtitles or the potentially erroneous words.

To reduce the subtitle content to essential information there have been approaches focussing on word frequency and cohesion or POS (Moran, 2012; Yang et al., 2010). Moran (2012) experimented with the replacement of low-frequency words with more frequent words as well as replacing words to obtain a higher cohesion between words, which was a kind of abstractive simplification approach. Yang et al. (2010) however employed more extractive techniques to show only keyword captions of words or nouns. Ferdiansyah and Nakagawa (2013) explored the use of inter- and intralingual captions as well as captions of important phrases and keywords to help foreign language learners, unfortunately they do not provide any specifics on how they selected the important phrases. Another approach Partial Synchronized Captions by Mirzaei et al. (2017) also investigated the use of partial captions to help second language learners, by synchronizing the captions on word level. They present a selection of difficult words for beginners, i.e. words with high speech rate, low frequency or academic terms. We, like them also use TED talks [3] to evaluate our approach. Our approach is also a method to distil the important content of the subtitles, but in contrast to the methods above, we want to

[3] https://www.ted.com/

explore an approach beyond fixed metrics and see whether a neural network can inherently learn such metrics by looking at training data.

3.2.2 Studies on the Effect of Partial Captions

The related studies regarding partial and keyword captions provide controversial results. Guillory (1998), Rooney (2014) and Mirzaei et al. (2017) focus their study on comprehension and listing and find positive effects of partial captions and no significant comprehension score differences compared to full captions. They argue based on the dual-coding theory that as they lower the input on the visual channel, the cognitive load has to be smaller.

Others, however, who investigated keyword captions and factored in participants subjective scores, report confusion and worse results than full captions or no captions (Montero Perez et al., 2014). Behroozizad and Majidi (2015) and Bensalem (2016) replicate the finding that keyword captions seem more a distraction than a comprehension aid for language learners.

This controversy is intriguing. In our user study, we want to tackle these questions and whether partial subtitles are enough to foster comprehension and how users perceive them in terms of cognitive load and helpfulness.

4 Our Neural Network Model

In this chapter the concept and implementation of the neural network model of this thesis is described. We present two architecture approaches discussed in Section 4.1, where also implementation details such as the modeling of the compression ratio are explained. For the implementation, we used Python frameworks and libraries mentioned in Section 4.2. To evaluate our model we conducted various experiments. The main results of those are reported in Section 4.3.

4.1 Architecture

We took the basic LSTM model by Filippova et al. (2015) as inspiration, however, the implementation is not exactly the same, as will be outlined in the following sections. Basically, we experimented with two basic architecture approaches: One architecture we call the *Simple-LSTM* architecture and one encoder-decoder architecture, which should help to model the desired compression ratio. We then varied different input features within these architectures to investigate the effect of those features on the compression results. Namely those input features were: POS tags of a sentence, the previously predicted label of a word and the target compression ratio of the sentence.

4.1.1 Simple-LSTM

The architecture of the *Simple-LSTM* is depicted in Figure 4, which is similar to the architecture proposed by Filippova et al. (2015), except that we use only one bidirectional instead of three stacked LSTM layers.

In the most basic version of *Simple-LSTM*, *Simple-LSTM_plain* we only use the sentence as input. This sentence is first transformed into a sequence of indexes, before being past to the model. These indexes are retrieved from the

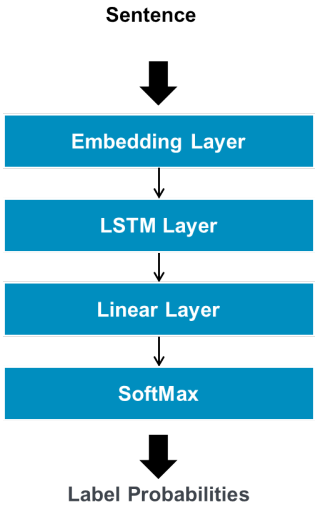

Figure 4: The architecture of the *Simple-LSTM* model without the additional embedding layer for POS.

constructed dictionary (i.e.word-id mappings) for the corpus. In the model, the words of the sentences are turned into word embeddings in the embedding layer. Then these word embeddings are passed to the LSTM layer which transforms the word embeddings into a sequence of hidden states. These hidden states are then mapped to the label space (the labels being 0 for DELETE and 1 for KEEP respectively). Finally a SoftMax layer is responsible for classification and outputs the label probabilities for each word. From these, the deletion sequence can be inferred by taking the maximum from each of the two label probabilities per word.

On top of *Simple-LSTM_plain* we build variants of that architecture with additional input features. *Simple-LSTM_POS* also uses the POS tags as input. Those were also turned into POS embeddings in a separate embedding layer. Together with the word embeddings fed into the LSTM layer for further processing, see Figure 5 exemplary for one sentence with four words, each word / POS tag denoted field in the row vector. Note that our notation uses row instead of column vectors, following the modelling of vectors (or tensors, which are a more general form of vectors matrices as they can have multiple

dimensions) in Pytorch, the framework presented in Section 4.2.

The compression ratio is given as an additional input feature to the model *Simple-LSTM_ compression*. Herefore, the compression ratios put into one of ten compression ratio bins, which in turn is represented as a ten-dimensional one-hot vector. A short example : Suppose, our desired compression ratio is 0.43, then the resulting compression ratio bin would be `[0.4;0.5)` which is represented by the `[0 0 0 0 1 0 0 0 0 0]` one-hot vector. The compression ratio vectors, together with the LSTM output are the input for the linear layer which transfers this information to the label space for the SoftMax layer to classify (c.f. Figure 6, here the POS embeddings are considered as well as inputs to the LSTM layer).

Simple-LSTM_ previous processes the output of the LSTM layer one by one to take into account the previously predicted label in the estimation of the target label of the current word. Hereby, the linear layer and the SoftMax layer process the token word by word. The previous label could be one of the following: `<SOS>`, denoting the start of the sentence, `1` for KEEP and `0` for DELETE. After each computation step the previous label is stored to be used in the computation of the next word. The processing of the previous label is also visualized in Figure 7. In this previous token processing pipeline, however, POS and compression ratio are considered as well.

We further implemented combinations of the above described models, namely:

- *Simple-LSTM_ POS_ previous*, which takes POS tags as additional input and the linear layer processes the LSTM output one by one.
- *Simple-LSTM_ POS_ compression*, again takes POS tags as further input and also uses the compression ratio vectors to influence the output of the linear layer.
- *Simple-LSTM_ previous_ compression* processes LSTM layer outputs one by one and also takes into account the compression ratios.

- *Simple-LSTM_POS_previous_compression* is a combination of all three basic concept mentioned above. Figures 5 to 7 can be considered as a visualization of its processing pipeline.

In sum, we present eight architectural variations of the *Simple-LSTM*, which we evaluate with different model parameters and corpus variations in Section 4.3.

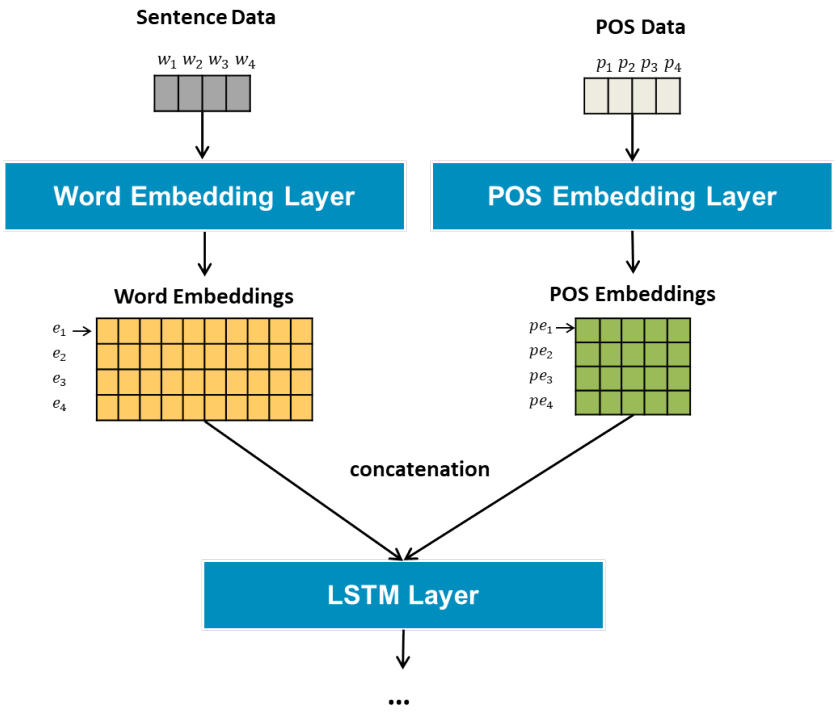

Figure 5: The processing pipeline for POS and word inputs (for a sentence with four words), i.e. a four dimensional row vector.The inputs are processed in separate embedding layers and the word embeddings (orange) and the POS embeddings (green) are then concatenated for further processing in the LSTM layer.

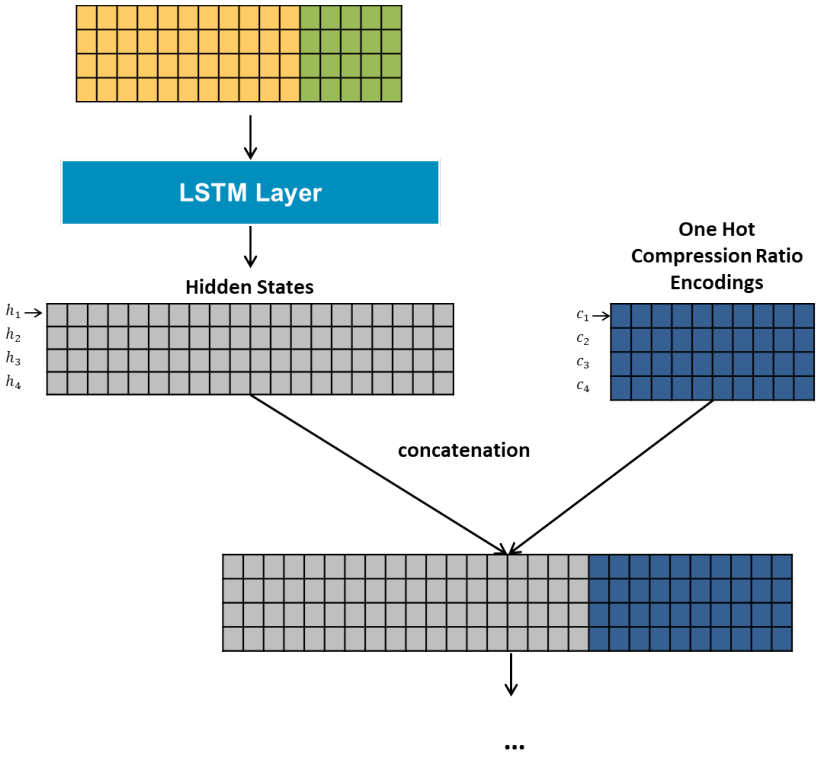

Figure 6: Combination of the hidden states (grey) with one-hot compression ratio vectors for further processing (dark blue).

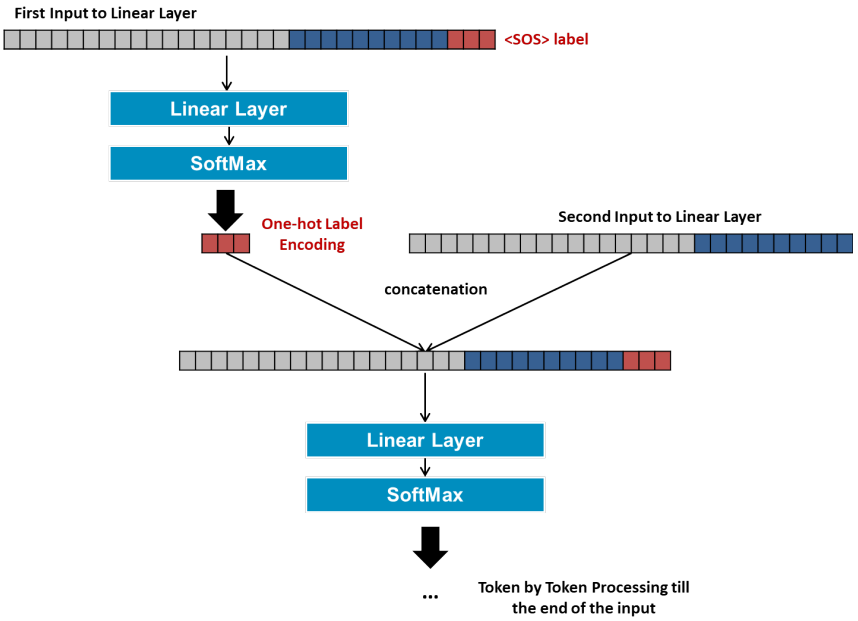

Figure 7: The processing of the hidden states (grey) which were combined with the compression ratios in 6 together with the previous labels (red), visualized for the first two inputs.

4.1.2 Encoder-Decoder Architecture

We wanted to investigate to what extend we could influence the resulting compression ratio with the compression ratio parameter and therefore also implemented an encoder-decoder or Seq2Seq architecture. Here, the input sentences are translated into a deletion sequence. In contrast to the standard encoder-decoder approach used for sentence translation, which has to cope with different lengths of input and output sequence, our framework has the benefit to deal with sequences of equal lengths, as the deletion sequence does not only model the words kept in the compression but also the ones to be deleted. Thus, every input word is also reflected in resulting sequence of the decoder, which makes input and output sequence of equal length. Encoders for sentence translation often read the input sentence backwards (Sutskever et al., 2014), we, however, use a bi-LSTM encoder and decoder as did Lai et al. (2017). As a classifier, we keep using SoftMax.

By employing an Seq2Seq architecture combined with the compression ratio parameter we hope to achieve higher compression ratio accuracy in relation to a desired target compression ratio and thus being able to control the length of the resulting compression to some extent.

Our encoder is modelled similar to the Simple-LSTM except that it does not contain the linear layer and the SoftMax layer. We again developed two variations of the encoder. One variation is using just the input sentence and one is also utilizing its POS tags. The decoder consists of one single bi-directional LSTM layer, followed by the linear layer and the SoftMax layer. The output of the LSTM is concatenated with the compression ratio and the previously predicted label, analogously to the process in the model *Simple-LSTM_previous_compression*. The layer structure of the general Seq2Seq model can be seen in Figure 8.

Depending on the encoder used, we get two Seq2Seq model configurations:

- *Seq2Seq-LSTM_previous_compression*, the model with the "plain" encoder not taking into account the POS features and the decoder using

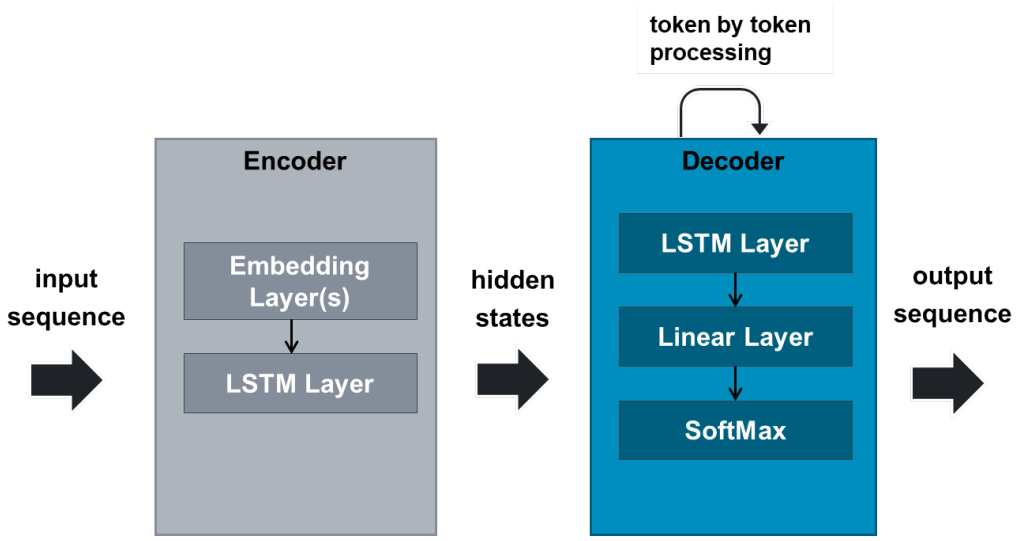

Figure 8: The architecture of the Seq2Seq model without the additional embedding layer for POS.

the previously predicted token and the compression ratio.

- *Seq2Seq-LSTM_POS_previous_compression*, where the encoder does look at the POS tags as well and the decoder is the same as described above.

4.2 Implementation Frameworks and Tools

We used the Python framework *PyTorch*[4] (version 0.40). The framework is basically a front end wrapper for the torch engine[5], which is an engine providing functionality for machine learning. *PyTorch*, in contrast to other frameworks like *TensorFlow* [6], is a dynamic framework. Tensorflow is a "define-compile-run" framework, requiring computation graphs, which are compiled and run. In PyTorch no such intermediate step is needed, you can simply

[4]https://pytorch.org/
[5]http://torch.ch/
[6]https://www.tensorflow.org/

write code and run it. This characteristic makes it easy to debug and intuitive to use.

Further, Pytorch is able to run processes on the Graphical Processing Unit (GPU) and supports parallel processing (Ketkar, 2017). One disadvantage, however, is that at the time of this thesis, it was a "young" framework still in beta. This signified some missing functionalities and changes in the *PyTorch* Framework during the development of this thesis, which is why the author decided to develop for one version of *PyTorch* available at the time to have some consistency in the code functionality. Despite its beta nature, PyTorch is well documented and already provides community support through forums.

Another library that was used was *gensim*[7].We used it to pre-train the word embeddings of the used dataset, with their implementation of the Word2Vec model (Řehůřek and Sojka, 2010).

4.3 Experiments

In this subsection the experiments are discussed. On the one hand, we evaluate our model configurations and architectures in terms of accuracy and F1-score, on the other hand we look a bit closer at the effect of the compression ratio parameter and its effect. Further, we report observations with data from subtitle files which we used for the user study described in 5.

Section 4.3.1 describes the data used for training and evaluation, Section 4.3.2 presents the model configurations, Section 4.3.3 explains the training process and finally Section 4.3.4 shows and discusses the results.

4.3.1 Datasets and Data Preparation

As datasets for training, development and evaluation we use the Google datasets for sentence compression[8]. It is a parallel corpus for sentence compression

[7]https://radimrehurek.com/gensim/

[8]https://github.com/google-research-datasets/sentence-compression

consisting of sentence-compression-pairs as described by Filippova and Altun (2013).

This dataset is based on news data obtained by a news crawler. More precisely, Filippova and Altun (2013) crawled the Google News site to extract news headlines and the first sentence of a news article. Headlines and sentences were preprocessed with a tokenizer, lemmatizer and a Part of Speech and Named Entity tagger. Further, they used a dependency parser to transform the sentences into dependency graphs. To create the compression, they use their tree-based compression algorithm Filippova and Strube (2008)

There are 200000 training and 10000 test instances provided by this dataset. We split the training set in development and training set and train only with 180000 instances and use 20000 as development set.

Listing 1 shows an example dataset entry (shortened for better presentation). It is formatted as a JSON object and consists of four nested objects, namely the `graph` object, the `compression` object, the `compression_untransformed` object and the `source_tree` object, as well as the three attributes `headline`, `compression_ratio` and `doc_id`.

```
{"graph": {
   ...
  },

  "compression": {
   ...
  },
  "headline": "Naked mole rats hold key to surviving stroke",
  "compression_ratio": 0.51999998,
  "doc_id": ...,
  "source_tree": {
    "id": "0",
    "sentence": "Researchers say that blind and almost hairless,
                 naked mole rats,
                 hold the key to surviving a stroke.",
    "node": [ ...
     {
      "form": "Researchers",
      "word": [ {
        "id": 8,
        "form": "Researchers",
        "stem": "researcher",
        "tag": "NNS"
      } ],
      ...]

      ...
  },

  "compression_untransformed": {
    "text": "Naked mole rats, hold the key to surviving a stroke.",
    "edge": [ {
      "parent_id": 18,
      "child_id": 16
      },
     ...]
    }
  }
}
```


Listing 1: Example Data Entry.

Important for construction of the training, development and evaluation data are the `source_tree` and the `compression_untransformed` object as well as the `compression_ratio`. As the model should not work on complex dependency tree structures but rather a simple sequence labelling approach, this untransformed data is used to extract the relevant information about the words of the sentence. We parsed the information nested in the source tree to obtain the words themselves, their parts of speech and word ids. The word ids are needed to parse the `compression_untransformed` object to get the target compression, as the `child_id` attribute in the `compression_-untransformed`'s `edge` object represented the word ids and therefore the words of the sentence present in the compression.

```
'original_sentence'= list(word_tuples(id,word,pos)),
'compression'= list(word_tuples(id,word,pos))
'deletion_sequence'=list(int) # only 0 or 1
'compression_ratio' = float # one of the ten compression ratio bins
```

Listing 2: Sentence object in pseudo code.

Additionally, the words of the sentence are filtered, special characters are cleaned and there are different normalization and cleaning options available. We constructed a dataset with and without punctuation characters as well as a dataset with only selected punctuation characters, which are commonly used to structure a sentence: `[,:;.!?]`.

Thus, for every dataset we have three variations further referred to as datasets `punct`, `no_punct` and `selected_punct`, with which we trained and evaluated. For each dataset and dataset configuration we created python lists with sentence objects, which we save in .pickle files so that we did not have to parse the data each time. An example of an sentence object is shown in Listing 2 in pseudo code. The target deletion sequence is calculated by comparing the words of the `original_sentence` and the `compression`.

To segment the data in batches and to transform it into tensors for training and evaluation, we implemented the `Dataset` and `DataLoader` class

provided by PyTorch[9] for the different input features required by the different model architectures. For indexing the words, POS tags and the compression ratio bins we created `word-id`, `tag-id` and `compression_ratio_lookup` files. The `DataLoader` then loads the saved lists of sentence objects and transforms them into batched input tensors for the models to use, relying on the previously specified Dataset structure and the given batch size parameter. Further, the `DataLoader` uses padding values to extend the input sentences all to the same length. The padding values later are masked in the network in order to not influence the calculations.

4.3.2 Model Configurations

For the experiments we want to test each of the ten model variations as described in section 4.1. Further we vary the punctuation characteristics of the dataset and train each model on each of the datasets. The third variation point is the dimensionality of the hidden states of the LSTM, for which we tested two values 120 as used by Filippova et al. (2015) and 256 as we wanted to test how the network would behave if their embedding dimensionality is equal to the number of hidden states. This leads to 60 different model configurations that are trained and evaluated, to which we refer to in the following notation later on: *architecture variation* `(punctuation characteristics, number of hidden states)`. For readibility reasons, however, the font highlighting is sometimes discarded.

The word embeddings were always pre-trained with *gensim*, which we configured as follows. We initialized the `sentences` parameters with our sentences from the training data set, the dimensionality of the word vectors was set to `EMBEDDING_DIM=256` and the minimum word occurrence count was set to five to simulate out-of-vocabulary (OOV) words. The pretrained weights from the Word2Vec model then are used to initialize the embedding layer of our models. The OOV embedding is defined as the average of all other word

[9] https://pytorch.org/docs/0.4.0/data.html?highlight=dataloader

vectors.

The other weights of the layers are initialized through Pytorch's implementation[10] of the uniform Glorot initialization (Glorot and Bengio, 2010), which assigns the uniform distribution $\mathcal{U} = (-a, a)$, where a is defined as

$$a = \text{gain} \times \sqrt{\frac{6}{\text{fan\_in} + \text{fan\_out}}}$$

where gain is an optional scaling factor set to one in our case and fan_in and fan_out are the number of input and output features.

4.3.3 Training Process and Parameters

Before training the model itself, we prepared the input data and we trained the gensim Word2Vec as described above. The batch size is set to 100 sentences each, and those batches are selected differently each epoch. We trained the models for ten epochs in total. After each epoch we evaluated the model on a sample of 10000 sentences from the training data, to see the training progress and also on the development dataset. A version of the model was saved after each epoch, as well as the sentence accuracy of the development data. At the end we selected the model of the with the best sentence accuracy on the development data to avoid overfitting.

Pytorch's implementation of NLL loss was used as a loss function and we used ADAM (Kingma and Ba, 2014) as an optimizer with a weight decay parameter of $1e^{-5}$. The training process is visualized in Figure 9.

[10] https://pytorch.org/docs/0.4.0/nn.html#torch-nn-init

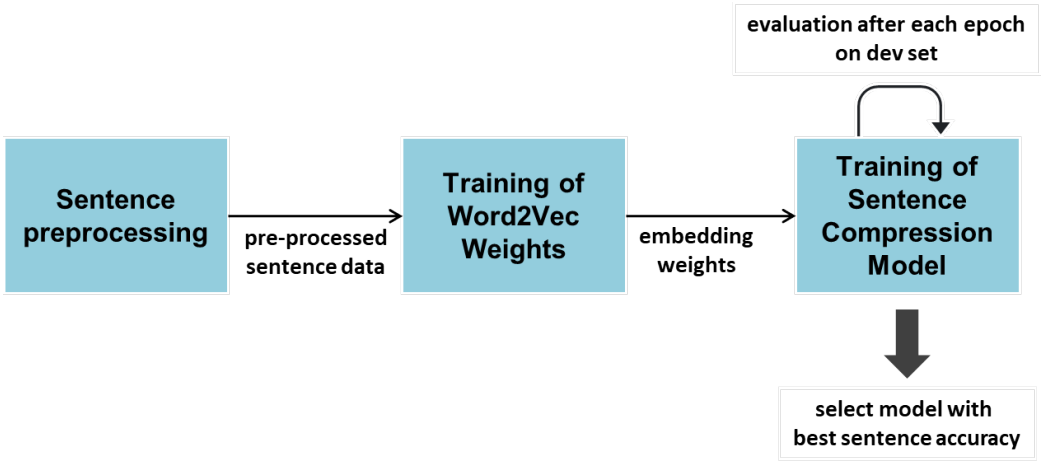

Figure 9: Training process with preparation steps.

4.3.4 Evaluation Results and Discussion

We report the sentence accuracy (A_s), token accuracy A_t, F1-score for KEEP ($f1_K$) and DELETE ($f1_D$) as well as the compression accuracy (A_c) for each model. For the compression ratio accuracy we checked whether the compression produced by the network adhered to the target compression ratio of the sentence, specified in the data.

While analysing the datasets, we noticed a problem with the distribution of the punctuation characters. In the development (in short: dev set) and training set, there was seldom to never a last punctuation character denoting the end of the sentence, while in the evaluation data set (in short: eval set), very often a sentence ended with a punctuation character. The distribution of punctuation characters is depicted in Table 1.

This led to no existing sentence accuracy, as the model did not learn to handle a last punctuation character (see Table 2, exemplarily shown for *Simple-LSTM_plain*. However, the other deletions seemed to be done almost accurately as the high per token accuracy indicated. Note that the development set is also not seen during training and receives good sentence accuracy scores in both scenarios. This pattern was reproduced over all 60 model con-

data set	**# sentences ending with punctuation character**
training	2
development	0
evaluation	9788

Table 1: Distribution of the punctuation characters in the datasets.

figurations.

model	A_s	A_t	A_c	$f1_K$	$f1_D$
development set (`punct`)	0.210	0.869	0.356	0.801	0.914
development set (`punct*`)	0.210	0.864	0.340	0.798	0.916
evaluation set (`punct`)	0.040	0.821	0.343	0.757	0.908
evaluation set (`punct*`)	0.191	0.850	0.330	0.791	0.914

Table 2: Performance with and without the last punctuation character.

Therefore, as sentence boundaries in spoken speech are not existent, we decided to remove the last punctuation character for evaluation, receiving datasets `punct*` and `selected_punct*` with the last punctuation characters removed. In the following we only report the results of those datasets and the `no_punct` dataset. Otherwise, we prepared the evaluation data with the same cleaning and parsing methods as the training data, i.e. if the training data was prepared with the `no_punct` option, the evaluation data was prepared with it too.

The within model configuration punctuation and number of hidden states, did have only a slight effect on the evaluation scores as can be seen in Table 3 for the *Simple-LSTM_plain* model. The lowest sentence accuracy is achieved by the *Simple-LSTM_plain*(`no_punct, 120`) with $A_s = 0.186$ and the highest by *Seq2Seq-LSTM_POS_previous_compression*(`no_punct,`

256) with $A_s = 0.327$. For the compression ratio accuracy the values range from $A_c = 0.299$ for *Simple-LSTM_plain*(no_punct, 120) to $A_s = 0.421$ for *Seq2Seq-LSTM_POS_previous_compression* (no_punct, 120). Table 4 and Table 5 show the best model configurations in terms of sentence accuracy and compression ratio accuracy.

model	A_s	A_t	A_c	$f1_K$	$f1_D$
Simple-LSTM_plain (no_punct, 120)	0.186	0.841	0.299	0.790	0.900
Simple-LSTM_plain (punct*, 120)	0.191	0.850	0.330	0.791	0.914
Simple-LSTM_plain (selected_punct*, 120)	0.199	0.850	0.330	0.790	0.922
Simple-LSTM_plain (no_punct, 256)	0.197	0.845	0.305	0.796	0.902
Simple-LSTM_plain (punct*, 256)	0.200	**0.854**	0.320	0.797	0.915
Simple-LSTM_plain (selected_punct*, 256)	**0.202**	**0.854**	**0.322**	**0.797**	0.912

Table 3: Results for all configurations of model *Simple-LSTM_plain* on the evaluation data set. The best scores are highlighted in bold.

Looking at the F1-scores, they remain similar throughout the model configurations and architectures. They cover values from $f1_K = 0.789$ (*Simple-LSTM_previous* (no_punct, 120)) and $f1_D = 0.880$ (*Simple-LSTM_previous_compression* configuration (selected_punct, 256)) to $f1_K = 0.859$ (*Seq2Seq-LSTM_POS_previous_compression* for all punctuation configurations with hidden 256 states) and $f1_D = 0.928$ for *Simple-LSTM_POS_previous*(punct, 120).

Token accuracy throughout the models is ranging from $A_t = 0.840$ for *Simple-LSTM_previous* configuration(no_punct, 120) to $A_t = 0.892$ for

model	A_s	A_t	A_c	$f1_K$	$f1_D$
Simple-LSTM_plain (selected_punct*, 256)	0.202	0.854	0.322	0.797	0.912
Simple-LSTM_POS (selected_punct*, 256)	0.224	0.859	0.330	0.806	0.920
Simple-LSTM_previous (selected_punct*, 256)	0.201	0.851	0.326	0.794	0.917
Simple-LSTM_compression (punct*, 256)	0.242	0.869	0.360	0.880	0.834
Simple-LSTM_POS_previous (selected_punct*, 256)	0.233	0.859	0.326	0.804	0.926
Simple-LSTM_POS_compression (punct*, 256)	0.263	0.876	0.383	0.840	0.891
Simple-LSTM_previous_compression (no_punct, 256)	0.248	0.866	0.398	0.829	0.900
Simple-LSTM_POS_previous_compression (selected_punct*, 256)	0.275	0.877	0.376	0.842	0.895
Seq2Seq-LSTM_previous_compression (no_punct, 256)	0.301	0.876	0.411	0.843	0.904
Seq2Seq-LSTM_POS_previous_compression (no_punct, 256)	**0.327**	0.888	0.410	0.859	0.911

Table 4: Best configurations in terms of sentence accuracy per model variation on the evaluation data set. The best sentence accuracy is depicted bold.

model	A_s	A_t	A_c	$f1_K$	$f1_D$
Simple-LSTM_plain (selected_punct*, 120)	0.199	0.850	0.330	0.790	0.922
Simple-LSTM_POS (selected_punct*, 256)	0.224	0.859	0.330	0.806	0.920
Simple-LSTM_previous (punct*, 256)	0.200	0.852	0.330	0.795	0.917
Simple-LSTM_compression (selected_punct*, 120)	0.238	0.866	0.410	0.828	0.905
Simple-LSTM_POS_previous (punct*, 256)	0.231	0.862	0.332	0.809	0.920
Simple-LSTM_POS_compression (no_punct, 256)	0.260	0.875	0.419	0.840	0.910
Simple-LSTM_previous_compression (no_punct, 256)	0.248	0.866	0.398	0.829	0.900
Simple-LSTM_POS_previous_compression (no_punct, 256)	0.266	0.872	0.419	0.83	0.918
Seq2Seq-LSTM_previous_compression (no_punct, 256)	0.301	0.876	0.411	0.843	0.904
Seq2Seq-LSTM_POS_previous_compression (no_punct, 120)	0.318	0.885	**0.421**	0.854	0.912

Table 5: Best configurations in terms of compression accuracy per model variation on the eval data set. The best achieved accuracy is bold.

Seq2Seq-LSTM_POS_previous_compression (punct*, 256).

To compare the different model variations and architectures, we also show the different scores of the configuration (no_punct, 256) of the different variations (see Table 6). The Seq2Seq achieve the highest sentence accuracy with $A_s = 0.301$ for *Seq2Seq-LSTM_previous_compression* and $A_s = 0.327$ for *Seq2Seq-LSTM_POS_previous_compression*.

model	A_s	A_t	A_c	$f1_K$	$f1_D$
Simple-LSTM_plain	0.197	0.845	0.305	0.796	0.902
Simple-LSTM_POS	0.212	0.853	0.307	0.806	0.909
Simple-LSTM_previous	0.195	0.842	0.300	0.790	0.909
Simple-LSTM_compression	0.234	0.870	0.379	0.833	0.887
Simple-LSTM_POS_previous	0.215	0.853	0.308	0.806	0.910
Simple-LSTM_POS_compression	0.260	0.875	0.419	0.840	0.910
Simple-LSTM_previous_compression	0.248	0.866	0.398	0.829	0.900
Simple-LSTM_POS_previous_compression	0.266	0.872	0.419	0.834	0.918
Seq2Seq-LSTM_previous_compression	0.301	0.876	0.411	0.843	0.904
Seq2Seq-LSTM_POS_previous_compression	**0.327**	**0.888**	0.410	**0.859**	**0.911**

Table 6: Models of the (no_punct, 256) configuration compared, results on evaluation dataset. Best values denoted in bold.

We further investigated the compression ratio parameter in two experiments with model *Seq2Seq-LSTM_previous_compression* (no_punct, 256). In the first experiment we looked at the accuracy of the different compression

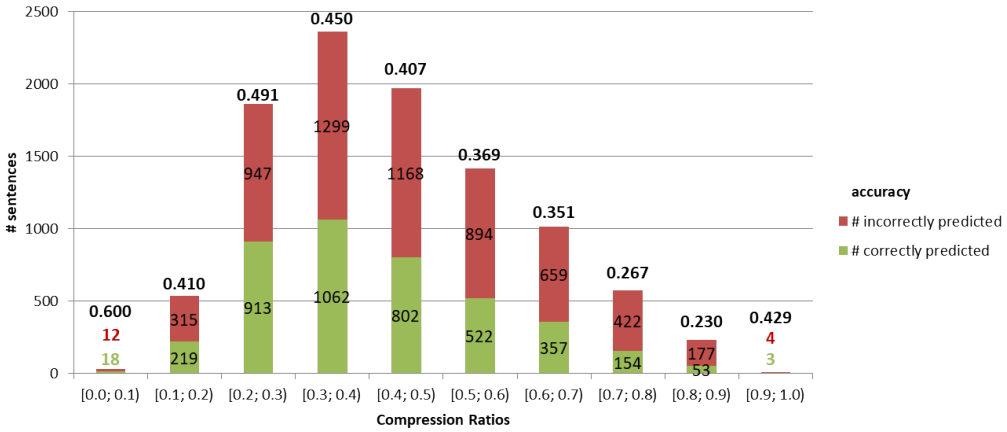

Figure 10: Analysis of the accuracy of the compressions according to their compression ratio classes in the evaluation data, tested on the model *Seq2Seq-LSTM_previous_compression* `(no_punct, 256)`.

ratio classes as denoted in Section 4.3.1,see Figure 10. So the accuracy here is defined by the correctly predicted members of that class in relation to the total amount of the members of the specific class.

In the second experiment we tried to compress sentences independently from their target label specified in the eval data, and gave all evaluation instances the same target compression ratio to investigate whether we could control the model to produce a compression at a specific compression ratio. By this setup we wanted to limit the influence of specific sentence structures that potentially could be corresponding to one specific compression ratio in the data. So, the goal was to isolate the effect of the compression ratio to some extent. The results are shown in Figure 11. In contrast to the experiment above, the correctly predicted compression ratios refer to the whole evaluation dataset.

For the user study we prepared compressions of spoken subtitles, which could be considered as out-of-domain data, as the model was trained on written language. We generated compressions for 204 sentences.The average OOV count was: $M_{OOV} = 26.3(\text{SD} = 9.81)$ We decided to discard punctuation, as

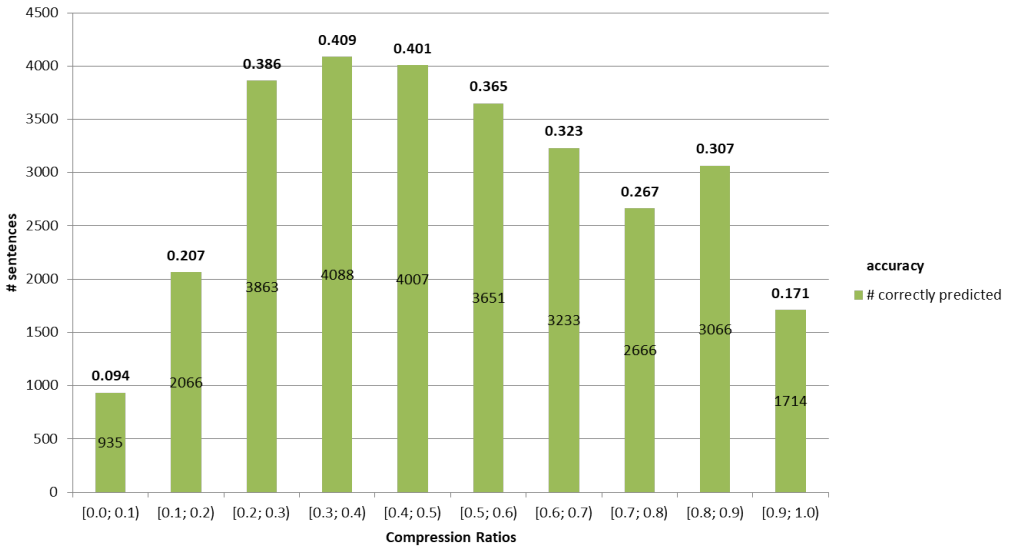

Figure 11: Analysis of the accuracy of the compressions when assigning one specific compression ratio class for all sentences in the evaluation data, tested on the model *Seq2Seq-LSTM_previous_compression* (`no_punct, 256`). The correctly predicted compression ratios refer to the whole evaluation dataset.

spoken speech is not structured like written speech and is missing sentence boundaries (Zhang et al., 2010). Also, as the data was not already available POS-tagged and we did not want to include extra noise with potentially false POS tags, we used the *Seq2Seq-LSTM_previous_compression* `(no_punct, 256)`, which does take plain sentences as input and was trained on data without punctuation. The target compression ratio parameter was set to the compression ratio class `[0.5;0.6)`. The resulting compression accuracy was 0.25, meaning 51 of the 204 sentences were compressed to a compression ratio in the class of `[0.5;0.6)`. Looking at the resulting compressions, sometimes they were grammatically correct, however there were some cases where the content was not reflected accordingly:

> Original sentence: The understanding of such molecular processes offers a panel of potential molecules that can be used to create novel anti-fungal treatments
>
> Compression: The understanding of molecular processes offers a panel of potential molecules **(GOOD)**
>
> Original sentence: Just seeing it there made people feel better and that was the most surprising thing
>
> Compression: there made people feel better and the thing **(POOR)**

To sum it up, the goals of the evaluation were threefold:

1. We wanted to test the performance of the different architectures and configurations.
2. The influence of the compression ratio parameter should be investigated.
3. The model's performance on subtitle data should be observed.

As noted above, the change of internal parameters seems only to have a slight effect on performance in our experiments. Models that rely on the

plain input or only the previous token, seem to be more sensible for these changes and punctuation and more hidden states seem to improve performance slightly. The Seq2Seq models seem not to rely on punctuation so much and in fact achieve better sentence accuracy without them. Token accuracy, however, is also better with punctuation for the Seq2Seq models. It could be that the punctuation characteristics are helpful at token-level, but for the higher level structures the model needs to relies on other inputs as well. These, however, are mere speculations on the author's part and more experiments are required to investigate that effect. Some slight variations could also be caused by the individual pre-trained embedding weights for each model.

For the F1-scores the DELETE scores are slightly higher than the KEEP scores, which could be caused by the fact that the training data contained more DELETE labels than KEEP labels and thus the model could be slightly better in learning to delete. This result, the effect of the last punctuation character described above and the sometimes poor performance on the out-of-domain data nicely depict the weakness of supervised models which heavily rely on the patterns seen in their training data. This problem is also recognized by others and Wang et al. (2017) propose the incorporation of syntactic constraints to improve out-of-domain performance, an approach to look into in the future.

It can be noted that the *compression_ratio* parameter seems to improve sentence accuracy and compression ratio accuracy, which naturally are related to some extent. Results of the compression ratio tests also indicate that the resulting length of the compressed sentence can be partly controlled by the compression ratio parameter, however, it seems that the length of the compression could also be influenced by other patterns of the original sentence, which the model learns. Thus, the model cannot yet generate compressions with arbitrary compression ratios for the same sentences. To achieve this, further adaptation of the model to specific compression ratios, through additionally training it with compression data adhering to those compressions ratios might be needed. This could result in better compression ratio

performance for that compression ratio bin and one would have 10 specialized models for each compression ratio bin. When needing a compression with a specific ratio, one would have to feed the sentence to the respective model. Yet, it remains to be seen whether there exists a sensible compression at each ratio for a sentence in terms of grammaticality and entailment. Further mechanisms to ensure entailment and grammaticality should be employed to avoid a too large trade-off between arbitrary compression ratio accuracy and readability and meaning preservation of a sentence. This matter should be investigated further as well.

When comparing our models to state of the art models listed in the Table 7, our model achieves good results, considering we use less data and input features, especially our F1-score seems to outperform the other models in their specified training/test configuration. Regarding the scores of the models from related work, it has to be noted, that we did not reimplement their approaches and their results are from the selected papers. Unfortunately they all used different amounts of training data and evaluation data, as well as different ways of data preparation which makes direct comparison difficult. Also in the case of Lu et al. (2017) it is not clear, which model of Filippova et al. (2015) they use as baseline.

Model	**# training / test**	A_s	$f1_K$
Filippova et al. (2015) LSTM	2 million / 1000	0.300	0.800
Filippova et al. (2015) PAR+PRES	2 million / 1000	0.340	0.820
Klerke et al. (2016) Best Scoring Model	8000 / 1000	-	0.810
Tran et al. (2016) Baseline	8000 / 1000	0.200	0.743
Tran et al. (2016) Best Scoring Model A_s	8000 / 1000	0.340	0.760
Tran et al. (2016) Best Scoring Model $f1_K$	8000 / 1000	0.320	0.770
Baseline Andor et al. (2016) (PAR+PRES)	2.3 million / 1600	**0.354**	0.828
Lai et al. (2017) Best Scoring Model	8000 / 1000	-	0.786
Lu et al. (2017) Baseline	200 000 / 10 000	0.232	0.757
Lu et al. (2017) Best Scoring Model	200 000 / 10 000	0.325	0.800
Seq2Seq-LSTM_POS_previous_compression (no_punct, 256)	180 000 / 10 000	0.327	**0.859**
Seq2Seq-LSTM_POS_previous_compression (punct*, 256)	200 000 / 10 000	0.316	**0.859**

Table 7: Our models compared to state of the art approaches. "Baseline" refers to the respective implementation of Filippova et al. (2015). Best scores are again denoted bold.

5 User Study

To evaluate the compressions constructed by the neural network model and to explore the effect of simplified subtitles, we conducted a user study. This section addresses the design and the results of the study. The following research questions are targeted with the study:

RQ1: Compared to standard subtitles, what are the effects on cognitive load and comprehension?

- **RQ 1.1:** Are simplified subtitles sufficient as a comprehension aid?
- **RQ 1.2:** How is the cognitive load affected by the shortening of the subtitles?

RQ2: What is the perceived usefulness of the different subtitle conditions?

RQ3: Are there differences between human and system simplified subtitles regarding cognitive load, subjective feedback and comprehension?

While **RQ1** and **RQ2** aim at evaluating the general concept, **RQ3** focusses on the evaluation of the system on a more usage oriented manner than the technical evaluation done in Section 4.3.

5.1 Methodology

The design of the study was a within-subjects design, more precisely a repeated measures design and thus every participant experienced every condition. We were testing three conditions: Human compressed subtitles (*compressed_h*), system compressed subtitles (*compressed_s*) and standard subtitles as a baseline condition (*full_base*).

The study had 30 participants in total, which were recruited through university mailing lists and social media. All of the participants were university students. The participants were between 20 and 32 years old ($M = 25.06$ $SD = 3.46$). Three of the participants were (near) native speakers, 24 of them considered themselves fluent and three reported to have a good knowledge of English. Their mother tongues were varied, 14 of them were German native speakers, two reported their mothertongue to be English, while the remaining 14 gave another mother tongue (details are visualized in Table 8). The subjects' exposure to English content and their subtitle usage behaviour is listed in Table 9. The majority of the people seemed to view or listen to English content *"often"* to *"always"*, subtitles, however, were used *"rarely"* to *"sometimes"* by most participants.

German	**14**
English	**2**
Other	
Arabic	*1*
Bosnian	*1*
Spanish	*2*
Urdu	*1*
Russian	*2*
Turkish	*1*
French	*1*
French(Canadian)	*1*
Chinese	*3*
Hindi	*1*
	14

Table 8: Overview over the mothertongue of the participants.

	Watching English Content	Subtitle usage
Never	0	4
Rarely	0	11
Sometimes	4	10
Often	14	5
Always	12	0

Table 9: Exposure to English Content (Audio and Video) and Subtitle Usage of Participants

We opted for three videos per condition to minimize the confounding effect of the speakers, so all participants had to watch nine videos in total. All subtitle conditions were prepared for every video, so that the subtitle condition could be evaluated independently from the video itself. This resulted in three groups with different video-subtitle mappings to which participants were randomly assigned. The order of the videos itself was randomized throughout conditions to eliminate side effects of a fixed presentation order of the video. By choosing this design we strive to limit confounding influences and aim for a high internal validity of the results.

5.2 Apparatus

As video material we used short TED talks which are available under the Creative Commons License[11] (video links see appendix). The subtitles and videos we downloaded from the non-profit subtitling plattform Amara[12], which recruits volunteers to create subtitles and makes those subtitles available for the general public with the aim to make multimedia content more accessible.

[11]https://www.ted.com/

[12]https://amara.org/en/teams/ted/videos/

Figure 12: Screenshot of a subtitled video.

The videos were three to four and a half minutes long (M = 3 min 41s, SD = 0.02) and we assigned them the subtitle conditions in a manner that each condition had approximately eleven minutes video duration in total.

The *full_ base* subtitle files had from 15 to 27 number of sentences (M = 22.67, SD = 4.03) and from 456 to 717 words (M = 535, SD = 83.04). To generate the compressed subtitles for the condition *compressed_ s*, we extracted the plain text from the subtitle files, which were in the .srt format and used the model *Seq2Seq-LSTM_ previous_ compression* `(no_punct, 256)` as described in Section 4.3. The sentences differing from the target compression ratio we left unchanged and did not apply manual corrections.

For the human-compressed subtitles we asked one person who was not familiar with the system and the comprehension questions to mark the important parts of each sentence, so that approximately 50 percent of the sentence was retained. However, it was also permitted to leave out sentences completely. A screen shot of the subtitled video is seen in Figure 12

base_full	*compressed_h*	*compressed_s*
12 00:00:31,708 --> 00:00:33,720 The Middle Ages, you see a lot of monks 13 00:00:33,744 --> 00:00:37,700 that were wearing garments that were cape-like, with hoods attached, 14 00:00:37,724 --> 00:00:39,017 so therefore, "hoodies."	12 00:00:31,708 --> 00:00:33,720 ... Middle Ages ... a lot of monks 13 00:00:33,744 --> 00:00:37,700 ... were wearing garments that were cape-like... 14 00:00:37,724 --> 00:00:39,017 ... "hoodies."	12 00:00:31,708 --> 00:00:33,720 The Middle ... you see a lot of monks 13 00:00:33,744 --> 00:00:37,700 that were wearing garments that were cape-like with hoods attached 14 00:00:37,724 --> 00:00:39,017 ...

Table 10: Example of a sentence in the subtitle files from the different conditions. Line breaks were added only in this Table for presentation purposes.

As a survey tool we used *LimeSurvey*[13], an open source software tool to create and execute surveys. The questionnaires were viewed on laptops with the current Firefox [14] version at the time. Participants used headphones when listening to the videos.

5.3 Procedure

The study was conducted in a computer lab at the university, where three participants could take the study at the same time. The experiment was supervised in case of questions.

Before taking the study the participants were briefed about the study purpose and signed a form of consent. Then they were randomly assigned a questionnaire group. It was made sure that none of the participants were in the same questionnaire group when taking the study in the same time slot.

[13]https://www.limesurvey.org/

[14]https://www.mozilla.org/de/firefox/

The participants had to type in their assigned survey url and then start the study.

The first questions of the study were a set of demographic questions where the participants were asked to give their occupation, age, gender, mother tongue and English language proficiency (in terms of the Common European Framework of Reference for Languages (CEFR Levels (European Council)) and with labels provided for those not familiar with the framework). Furthermore, the participants were asked to state their exposure to English video content and their subtitle usage behaviour, answer options were a 5-point likert-scale ranging from *"1 - never"* to *"5 - always"* (Vagias, 2006).

After the demographic question block the viewing of the nine videos started and the participants were asked to use the headphones provided by the experimenters. Succeeding each video were statements regarding the cognitive load, based on the NASA TLX (Hart and Staveland, 1988), subjective feedback questions with 5-point likert-scale answer options ranging from *"1-disagree"* to *"5 - agree"* (Vagias, 2006) as well as three comprehension questions. The cognitive load questions and the subjective feedback questions can be seen in Table 11 and Table 12 respectively. The labels in brackets were not shown to the participant, they are rather for the reader's benefit, in order to know the abbreviations of the questions used in later sections.

Finally, we asked the participants for concluding feedback regarding their preferences of standard versus the abbreviated subtitles and the context in which they might want to use simplified subtitles. The whole survey structure can be seen in the appendix A.3.

***Mental demand**: How mentally demanding was it to read the subtitles?*
***Temporal demand**: How rushed did you feel when reading the subtitles?*
***Effort**: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?*
***Frustation**: How irritated by the subtitles were you when watching the video?*

Table 11: Cognitive Load Questions inspired by the cognitive load categories of NASA TLX.

The subtitles were easy to read.	**(s1: easy to read)**
The subtitles helped me to understand the content.	**(s2: helped to understand)**
The subtitles were confusing.	**(s3: confusing)**
The subtitles were too short.	**(s4: too short)**
The subtitles were too long.	**(s5: too long)**
The subtitles contained all important information.	**(s6: important information)**

Table 12: Subjective Feeback Questions.

5.4 Results

In the following we present the results of the user study. For statistical testing the software GraphPad Prism version 8.0.0 for Windows[15] was used.

5.4.1 Comprehension

The baseline standard subtitles had a mean of M = 7.07 with a standard deviation of SD = 1.48. For the *compressed_h* condition a mean of M = 6.67 was a achieved (SD = 1.52). The condition *compressed_s* yielded a mean of M = 6.77 (SD = 1.79) of nine possible correct answers per participant. A one-way repeated-measures ANOVA with a Geisser-Greenhouse correction did not yield any significant difference ($F(2, 29) = 0.601$, $p = 0.5506$).

5.4.2 Cognitive Load

In the following the results for the cognitive load categories **mental demand**, **temporal demand**, **effort** as well as **frustration** are reported. The results consist of overall scores (over all 90 answers) and over the aggregated answers per participant (as we counterbalanced for the effect of the single videos), i.e. over 30 answers. The overall descriptive statistics are visualized in Table 13.

To aggregate the data per participant we took the median answer. We compared the scores of the different conditions with repeated-measures Friedman tests ($\alpha = 0.05$) followed by Dunn's multiple comparisons test with p-value correction, taking into account the three multiple comparisons. The cognitive load scores of all categories over the aggregated data per participant is shown in Figure13 as box-plots. The number 1 refers to *"very low"* and 5 to *"very high"*.

[15] https://www.graphpad.com/

mental			
M	2.34	**2.44**	1.88
SD	**1.04**	1.03	0.78
median	**2.00**	**2.00**	**2.00**
mode	**2.00**	**2.00**	**2.00**
temporal			
M	1.73	1.87	**1.92**
SD	0.70	0.81	**0.85**
median	**2.00**	**2.00**	**2.00**
mode	**2.00**	**2.00**	**2.00**
effort			
M	2.56	**2.71**	2.01
SD	1.17	**1.19**	0.83
median	2.00	3.00	**2.00**
mode	**2.00**	**2.00**	**2.00**
frustration			
M	2.66	**2.84**	1.61
SD	1.24	**1.32**	0.80
median	3.00	3.00	**1.00**
mode	2.00	2.00	**1.00**

Table 13: The overall descriptive statistics for the cognitive load categories, the highest values denoted bold.

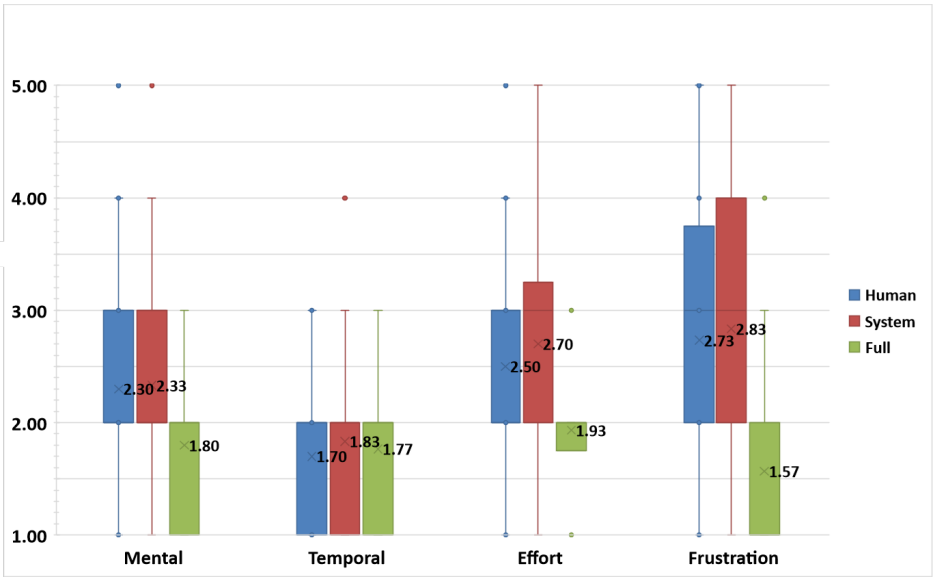

Figure 13: Overview of cognitive load based on the aggregated data per participant. The values at the denote the mean of the aggregated data.

The Friedman test did find a significant difference of the median answers between the subtitle conditions ($\chi^2(2) = 7.719$, $p = 0.0211$, $p < 0.05$) regarding the **mental demand**. The pairwise comparison, however, did not find any significant differences between the subtitle condition pairs: The rank sum difference (rsd) of $full\_base - compressed\_h$ was -12.50, but not significant with $p_{corrected} = 0.3197$. Likewise,the rank sum differences of $full\_base - compressed\_s$ and $compressed\_h - compressed\_s$ were -14.50 and -2.00, but also not significant with $p_{corrected} = 0.1836$ and $p_{corrected} > 0.99$.

There were significant differences between the conditions for **effort** category, according to a Friedman test ($\chi^2(2) = 12.87$, $p = 0.0016$, $p < 0.05$). The post-hoc test showed that the full subtitles required significantly less effort compared to the system-compressed subtitles($p_{corrected} = 0.021 < \alpha$). Between full subtitles and human-compressed subtitles and between human- and system-compressed subtitles, the difference was visible (rsd $= -16.50$ and rsd $= -4.50$), but not significant with $p_{corrected} = 0.0995$ and

$p_{corrected} > 0.99$. There was no significant difference between the median answers of the conditions regarding the **temporal demand** ($\chi^2(2) = 0.3158$, $p = 0.8539$).

However, the median answers of the conditions varied significantly in the aspect of **frustration** according to a Friedman test with $\chi^2(2) = 21.82$, $p < 0.0001$. The participants seemed to be significantly less irritated by the full subtitles compared to the system-compressed subtitles ($p_{corrected} = 0.0019$). Further, the full subtitles also caused less irritation than those of the *compressed_h* condition($p_{corrected} = 0.0090$). There was no significant difference between conditions *compressed_h* and *compressed_s*(rsd $= -3.50$ and $p_{corrected} > 0.99$).

5.4.3 Subjective Scores

The presentation of the subjective scores is analogous to the presentation of the cognitive scores stated above, consisting of overall results based on the total number of answers (shown in Table 14 and Table 15) and as well inferential statistics relying on the aggregated answers per participants. To aggregate the data per participant we once more took the median answer. We compared the aggregated scores of the different conditions again with repeated-measures Friedman tests ($\alpha = 0.05$) followed by Dunn's multiple comparisons tests with p-value correction.

The median ratings of the first subjective question **s1: easy to read** are significantly regarding the different subtitle types according to a Friedman test ($\chi^2(2) = 16.32$ and $p = 0.0003$). The Dunn's post-hoc test showed pairwise significances between *full_base* and *compressed_h* ($p_{corrected} = 0.0425$) and between *full_base* and *compressed_s* ($p_{corrected} = 0.0090$). There was no significant distinction between human- and system-compressed subtitles.

The overall distribution of **s2: helped to understand** is visualized in Figure 14. There were significant differences found between the three subtitle variations in the scores of this question ($\chi^2(2) = 21.68$ and $p < 0.0001$). A

pairwise-comparison with the Dunn's test affirmed that full subtitles were perceived as more helpful than system-compressed subtitles ($p_{corrected} = 0.0004$). Not significant were the differences between complete subtitles and human-compressed subtitles ($p_{corrected} = 0.0508$) as well as the differences between human-compressed and system-compressed subtitles ($p_{corrected} = 0.4467$).

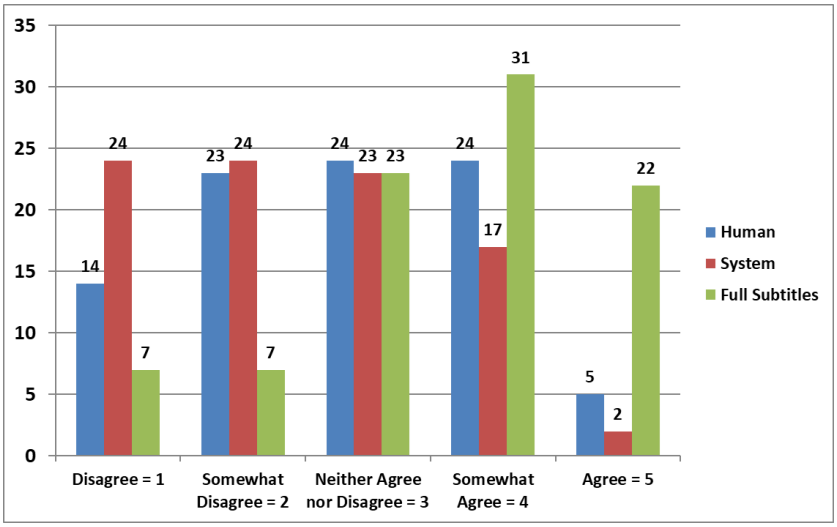

Figure 14: Overall scores of s2.

Furthermore, for question **s3: confusing** a Friedman test uncovered a significance in the difference of the median rankings between conditions with $\chi^2(2) = 38.21$ and $p < 0.0001$. Post-hoc tests detected that complete subtitles are regarded as significantly less confusing than those of the *compressed_h* ($p_{corrected} < 0.0001$) condition and also significantly less confusing than the subtitles compressed by the system ($p_{corrected} < 0.0001$). No further pairwise significances were found.

	compressed_h	compressed_s	full_base
s1: easy to read			
M	3.68	3.38	**4.37**
SD	**1.23**	1.13	0.84
median	4.00	4.00	**5.00**
mode	4.00	4.00	**5.00**
s2: helped to understand			
M	2.81	2.43	**3.60**
SD	1.15	1.14	**1.16**
median	3.00	2.00	**4.00**
mode	3.00	1.00	**4.00**
s3: confusing			
M	3.01	**3.26**	1.48
SD	1.22	**1.23**	0.83
median	3.00	**3.50**	1.00
mode	2.00	**4.00**	1.00

Table 14: Descriptive statistics of questions s1 to s3.

Again, a Friedman test exposed significant differences among the subtitle conditions ($\chi^2(2) = 38.21$ and $p < 0.0001$) regarding question **s4: too short**. The pairwise differences of *full_base* and *compressed_h*, as well as those between *full_base* and *compressed_s* were determined as significant with Dunn's test with $p_{corrected} < 0.0001$ for each pair-wise comparison. No significance, however, was detected between the *compressed_h* and *compressed_s* condition. As well, no significant differences were found between the conditions for **s5: too long** ($\chi^2(2) = 5.450$, $p = 0.655$).

Regarding the inquiry whether the subtitles of the respective conditions contained the relevant information (c.f. **s6: important information** a Friedman test indicated significant differences of the median ratings per participant between the conditions ($\chi^2(2) = 51.86$ and $p < 0.0001$). The Dunn's post-hoc test revealed that the baseline subtitles were perceived as significantly more reliable in transmitting all the important information compared to *compressed_h* ($p_{corrected} < 0.0001$) and *compressed_s* ($p_{corrected} < 0.0001$). There was no significant difference found between *compressed_h* and *compressed_s* ($p_{corrected} = 0.3197$) though some small difference is visible (rsd = 12.50).

	compressed_h	*compressed_s*	*full_base*
s4: too short			
M	3.32	**3.63**	1.43
SD	**1.40**	1.28	0.75
median	**4.00**	**4.00**	1.00
mode	**4.00**	**4.00**	1.00
s5: too long			
M	1.48	1.54	**1.90**
SD	0.70	0.83	**1.08**
median	**1.00**	**1.00**	**1.00**
mode	**1.00**	**1.00**	**1.00**
s6: important information			
M	2.79	2.03	**4.72**
SD	**1.35**	1.20	0.77
median	2.00	2.00	**5.00**
mode	4.00	1.00	**5.00**

Table 15: Descriptive statistics of questions s4 to s6, the highest values in bold.

5.4.4 Concluding questions

Participants seem to prefer the complete subtitles with a mean of M=4.43 (SD = 1.12 , median = 3, mode = 5). Shortened subtitles got ratings of a mean of M=1.80 (SD = 1.05 , median = 1,mode = 1).

Lectures and talks received the most votes on the question: "For which content would you like to have shortened subtitles?" with 12 and 10 votes respectively. For this question participants could select multiple options provided. On the "Other" category, documentaries, speeches or *"easy to understand content"*, opposed to *"content [featuring] people with strong accents"*. Six people also mentioned *"None"*. The results are visualized in Figure 15.

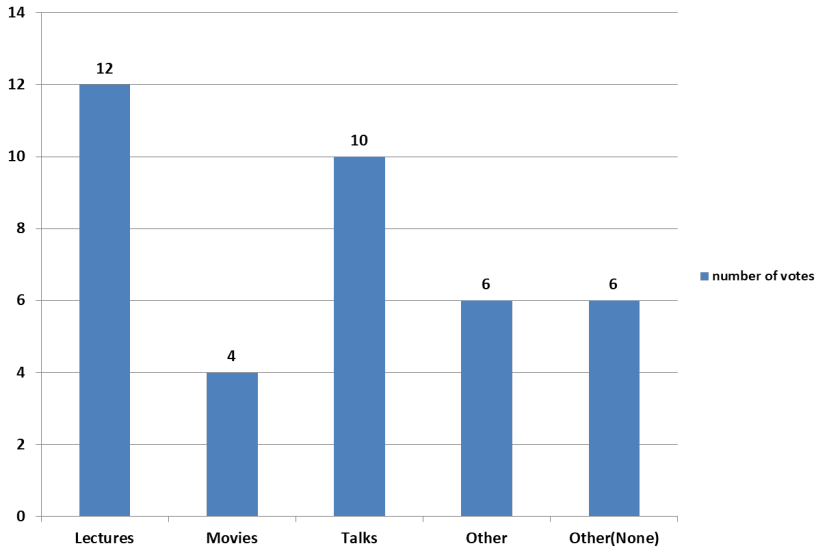

Figure 15: Results to *"I would like to have subtitles for the following content"*.

5.5 Limitations

Due to the small sample of the study, the results presented here are not generalizable. Further, the English level of the participants could have had an confounding influence on the results as well as their subtitle usage behaviour. Regarding the methodology of the comprehension test, multiple choice questions are prone to guessing and might not need a deeper comprehension of the content (Basaraba et al., 2013). We tried to limit guessing by providing an "I don't know option" and encouraged them to use it, if they failed to know an answer. Also, it shall be noted that the measured cognitive load is subjective and future studies should maybe measure the cognitive load directly, i.e. through eye tracking data like pupil dilation as done by Kruger et al. (2013). We also did not control how the participants viewed the video, some used full-screen mode others did not. This could also have an impact on the results. Additionally, some participants used the possibility to rewind, our measured viewing times, however, showed that there is no strong effect of that visible in the data.

6 Discussion

Here, the results of the user study are discussed, to state the findings of the user study and to draw consequences for future developments of the prototype.

6.1 Findings of the User Study

Regarding **RQ 1.1**, one could cautiously conclude that the lack of significant differences between the subtitle conditions is an indication that 50 % of the subtitle content is already enough for comprehension, which would replicate the findings of Rooney (2014) as well as Guillory (1998).

For the cognitive load (**RQ 1.2**), however, we could affirm the findings of Montero Perez et al. (2014), Behroozizad and Majidi (2015) and Bensalem (2016). Partial subtitles are a source of confusion, especially seen in the frustration and effort scores, slightly less so for the human-compressed subtitles. This could be explained by a lack of *"belongingness"* (c.f. Grimes (1991)) of the two stimuli. Audio and subtitles differ too much, so that they could not be processed together and instead compete for attention, thus resulting in higher frustration and effort after the theory of Grimes (1991). A solution to that problem would be to alter the compression in so far, that it is shorter but still can be aligned to what the participant is hearing. Suggestions for that will be given in the next section. The confusion is also reflected in the participants comments, e.g.:

> *P1: "Having words left out in subtitles causes a mental context change, resulting in me suddenly having to focus especially hard on the sound and being frustrated if I couldn't understand it."*

> *P2: "For People that have a good grasp on the English language it seems more annoying than useful, since you concentrate on the*

missing words and contexts in the subtitles more than what is actually said in there."

P3: "...it distracts me from the talk and I am not totally focused so I tend to miss information because I am more thinking about the subtitles than the actual speech"

The perceived usefulness (**RQ2**) of the compressed subtitles was mixed and there was a strong tendency towards full subtitles or shortened subtitles with different content. Though the full subtitles received a significantly higher score regarding the readability, it has to be noted that the overall modes in the data suggest that compressed subtitles were not hard to read as well. It could be that participants just felt more comfortable reading full subtitles because they were used to this subtitle type, as one participant stated:

"...I am used to reading complete subtitles as that is my default setting."

This replicates the findings of Berke et al. (2017), who also observed that people tend to reject subtitle designs they are not used to and find them confusing.

The helpfulness of full subtitles was perceived as better as well, compared to the shortened subtitles, which could be closely related to the fact that they seemed to cause irritation among the participants. Additionally, participants in general did not seem to be disturbed by long subtitles, on the contrary, shortened subtitles suggested lack of information as the following comments show:

P4: "[the use of shortened subtitles] made me feel like I was missing things in certain videos, especially the videos about topics unfamiliar to me."

P4: "As a non native English speaker, I feel like if the subtitles are not complete (some words or entire chunks of text are missing)."

The parameter of length could however, be of importance, e.g. when the screen size is limited. This scenario should be investigated in future studies.

Regarding the comparison of human vs. system compressed subtitles no significances could be found and thus no clear conclusion can be drawn. Looking at the overall descriptive statistics the human compressed subtitles are often better in terms of median and modes, which could suggest that the idea in general could be helpful but our system is not mature enough yet to be of benefit. The evidence for that fact, however, is too weak to draw any conclusion at this stage and should be investigated again in the future. Yet some participants' comments show that they often do not reject the idea in general, but are critical towards the way the idea is implemented:

P6: "I like the Idea, but i find it irritating when, in my opinion, important details were excluded from the subtitles, like names, verbs to give the sentence its meaning and so on."

P7: "Most of the time not the easy words were left out but the interesting and hard to understand ones, although it should be the other way around."

In short, the simplified subtitles seemed to be sufficient for comprehension, but were not yet perceived much helpful by the participants. To solve this issue, one still has to work on the subtitle design and content creation, topics we will discuss in the next section.

6.2 Lessons learned

Though the feedback of the study is rather critical, there are a lot of aspects which are helpful for future development and design of compressed subtitles. We summarize them as design suggestions:

- **Know your data and usage context.** One reason on the system side for the production of confusing subtitles, is that for one it was not adapted for speech data. Speech is structured differently than written text and by itself lacks punctuation or the hierarchical structuring like headlines or paragraphs. Instead one could use prosodic information to determine importance (Zhang et al., 2010). Further, the model did not take into account the special line breaks of subtitles. It processed and compressed the sentences as a whole to compose a compression, which even might be readable when seen as a whole. However, if split into multiple parts, the compression might be of source for confusion, as it potentially combined sentence fragments from different parts of the sentence, which if seen standing on their own again do not make sense to the reader.

  This is closely related to the usage context. Reading subtitles is different to reading a static text, as one has no time to regress if something is confusing (Krejtz et al., 2016), so the grammaticality of the sentence or its subparts is even more important.

  One possible solution could be to include the line breaks of the subtitles, so that the system creates compressions of parts of the sentences and to further include some syntactical constraints to ensure grammaticality (c.f. Wang et al. (2017))

- **Take care with content selection.** Participants stressed the need of important keywords and would rather accept a system which perceivably selects the important facts in form of content words or numbers or named entities.

From the system side, this means to pay extra attention to an "importance" measure of the content, which could additionally be fed to the neural network to improve performance. This could be in form of additional tf-idf embeddings (c.f. Nallapati et al. (2016), named entity information or even gaze data like Klerke et al. (2016). Eye tracking data as well as phonological data could help to single out difficult parts or important parts of the content. Here the system could additionally be trained with those data to learn features as fixation points, pitch or speech rate.

- **Suggest informativeness rather then lack of information** We showed the participants dots to indicate something is missing. However, visualization of negative aspects such as missing information in our case, or potentially faulty words in ASR generated subtitles in the case of the study of Piquard-Kipffer et al. (2015), seem to elicit negative responses. Piquard-Kipffer et al. (2015) report a preference for "positive highlighting". So, one could explore highlighting the important parts. However, markup has to be used with care as the study of Berke et al. (2017) showed.

  Perhaps one should focus on more than just the length of the sentence, as shortening is only one part of the simplification process (Petersen and Ostendorf, 2007; Shardlow, 2014). One could further employ abstractive summarization or simplification methods to make the content more understandable, e.g replace words difficult for language learners with easier synonyms.

- **Minimize confusion.** The sentence fragments presented should be perceived in line with the audio and be understandable in themselves and should not elicit false information. The latter could be achieved by incorporating more syntactical information such as dependency parse trees in the training process of the model (Filippova et al., 2015). Furthermore, the concept of logical entailment could be utilized as well and

trained in a multitask-learning approach together with sentence compression as suggested by Guo et al. (2018). Further, one could train the model also on specific manual subtitle editing rules as described by Karamitroglou (1998) and provide according labelled training data to learn those patterns.

- **Design for individual needs.** Reading subtitles is dependent on someone's reading as well as language skills (Burnham et al., 2008) and thus has influence on their subjective reaction to subtitle prototypes. Further persons with hearing impairments have their own needs regarding the system design (Kawas et al., 2016). Furthermore, the participants expressed the wish for flexible systems adapting to their content and individual needs. Therefore, it might be sensible to provide either a variety of systems for different needs or work on a automatically adaptive system automatically learning the needs of the user. The latter approach is the more difficult one, for starters one could investigate the provision of various systems each tailored to the needs of the respective users. In respect to the language learners, this could mean systems doing paraphrasing and summarization by taking into account the language level and the according vocabulary. For achieving these kind of designs, it is crucial to incorporate the respective target group in the design process, e.g. by doing extensive user study with focus groups and diary studies (Kawas et al., 2016) or end user profiling (Matamala et al., 2018).

7 Conclusion and Future Work

Providing technologies to help to deal with an abundance of information or even helping to provide access to the latter is important in the age of ubiquitous information. In thesis, we tackled the case of spoken information and subtitles as assistive technology and investigated means to compress them to the essential information.

To summarize, we provide two contributions: the implementation of a neural network model for sentence compression as well as the evaluation of the concept of compressed subtitles in a user study.

The neural network model was tested with different configurations and achieved results comparable to state of the art approaches. We used a Seq2Seq architecture in combination with a compression ratio parameter to control the resulting compression ratio and received a compression ratio accuracy of 0.421 for the best-scoring model configuration. However, this model is not yet capable of producing arbitrary compressions of desired compression ratios for specific sentences, but could be used as baseline for future research in that direction.

Results of the user study show that shortened subtitles could be enough to foster comprehension, but result in higher cognitive load for the participants as audio and subtitles are perceived as conflicting rather than connected stimuli. Despite that critical feedback we believe the idea of simplified subtitles has potential and gathered design suggestions to improve future implementations in respect to their usability.

Future work thus should try to improve the model both in terms of the technical performance and the resulting usability of the results. One should further adapt the model for speech data to achieve better performance on subtitle data, by training with phonetic datasets as well or using subtitles as corpora for training. As well one could include additional information like gaze data or dependency structure to improve the relevance and coherence of the resulting compression (Filippova et al., 2015; Klerke et al., 2016).

Additionally one could employ the principle of multitask-learning to learn multiple tasks related to sentence compression or simplification in parallel (Guo et al., 2018; Klerke et al., 2016).

Besides improvements, one could extend the approach to different usage scenarios, for example reading in augmented reality, where screen space is limited and one needs to perceive other visual stimuli besides the subtitle text. Our subtitle compression in combination with ASR technologies augmented reality (AR) glasses could be an assistive device for hearing impaired people to make e.g. the content of lectures more accessible. There already exist approaches combining ASR and AR technologies (c.f. Mirzaei et al. (2014)). Others already investigated methodologies to facilitate reading in AR via comparing different text presentation modalities (Rzayev et al., 2018). However, they do not yet apply content simplification.

Our envisioned pipeline could be as follows. The spoken content could be first recognized by the ASR. The ASR transcriptions (perhaps together with the audio data) could be sent to our compression model. This calculates the resulting compression and sends it to a front end application on AR glasses of the user sitting in the lecture. This is only one of many application scenarios.

We believe our approach can be used as a starting point when investigating the use of deep learning systems as part of an assistive device for humans to support language understanding. However, the results of our user study show, that in this case, mere technical evaluation is not enough. When designing systems for the user, you have to design it with the user. In our opinion neural sentence simplification and compression holds great potential to make "the magic of words" more accessible and this potential should be further explored in the future.

A User Study Resources

A.1 Video Resources

https://www.youtube.com/watch?v=qpfq3xCdAu4
How fungi recognize (and infect) plants | Mennat El Ghalid

https://www.youtube.com/watch?v=uv5-hIif7BQ
A rare galaxy that's challenging our understanding of the universe | Burçin Mutlu-Pakdil

https://www.youtube.com/watch?v=TRQdHrGuVgI Could a Saturn moon harbor life? - Carolyn Porco

https://www.youtube.com/watch?v=EaY_6muHSSI Finding planets around other stars | Lucianne Walkowicz

https://www.youtube.com/watch?v=YX_OxBfsvbk Why is 'x' the unknown? | Terry Moorel

https://www.youtube.com/watch?v=IBf9pXOmpFw Why the pencil is perfect | Small Thing Big Idea, a TED series

https://www.youtube.com/watch?v=xqzLm0Xua8g The 3,000-year history of the hoodie | Small Thing Big Idea, a TED series

https://www.youtube.com/watch?v=3Va3oY8pfSI How the hyperlink changed everything | Small Thing Big Idea, a TED series

A.2 Example Subtitle Files

```
1
00:00:00,825 --> 00:00:03,515
"Will the blight end the chestnut?

2
00:00:03,857 --> 00:00:05,857
The farmers rather guess not.

3
00:00:05,881 --> 00:00:08,119
It keeps smouldering at the roots

4
00:00:08,143 --> 00:00:10,151
And sending up new shoots

5
00:00:10,175 --> 00:00:11,841
Till another parasite

6
00:00:11,865 --> 00:00:14,269
Shall come to end the blight."

7
00:00:16,510 --> 00:00:18,549
... beginning ... 20th century,

8
00:00:18,573 --> 00:00:23,220
... eastern American chestnut population counting ... four billion trees,

9
00:00:23,244 --> 00:00:26,345
was ... decimated ... fungal infection.

10
00:00:26,369 --> 00:00:29,577
Fungi ... most destructive pathogens of plants,
```

```
11
00:00:29,601 --> 00:00:32,323
including crops ...

12
00:00:32,673 --> 00:00:34,234
... imagine ... today,

13
00:00:34,258 --> 00:00:37,125
crop losses ... fungal infection

14
00:00:37,149 --> 00:00:41,065
are estimated at billions of dollars ...

15
00:00:41,585 --> 00:00:45,283
... represents enough food ... half a billion people.

16
00:00:45,609 --> 00:00:47,967
... severe repercussions,

17
00:00:47,991 --> 00:00:51,411
... famine ...

18
00:00:51,435 --> 00:00:54,839
... reduction ... for farmers ... distributors,

19
00:00:54,863 --> 00:00:56,791
high prices ...

20
00:00:56,815 --> 00:01:01,539
... risk of exposure to mycotoxin,
```

```
21
00:01:02,318 --> 00:01:03,580
.. problems ...

22
00:01:03,604 --> 00:01:06,321
... current method ... to prevent ... treat

23
00:01:06,345 --> 00:01:07,887
... diseases,

24
00:01:07,911 --> 00:01:12,220
... genetic control, exploiting natural sources of resistance,

25
00:01:12,244 --> 00:01:15,625
crop rotation ... seed treatment...

26
00:01:15,649 --> 00:01:18,331
... limited ... ephemeral.

27
00:01:18,879 --> 00:01:21,299
They have to be constantly renewed.

28
00:01:21,323 --> 00:01:25,577
... need to develop ... efficient strategies

29
00:01:25,601 --> 00:01:30,720
... research ... to identify biological mechanisms

30
00:01:30,744 --> 00:01:34,410
... targeted by ... antifungal treatments.

31
```

00:01:37,529 --> 00:01:40,663
... fungi ... cannot move

32
00:01:40,687 --> 00:01:44,212
... only grow by extension to form a ... network,

33
00:01:44,236 --> 00:01:45,386
the mycelium.

34
00:01:46,284 --> 00:01:50,537
... Anton de Bary ...

35
00:01:50,561 --> 00:01:54,117
....presume ... fungi ... guided by signals

36
00:01:54,141 --> 00:01:56,077
... from the host plant,

37
00:01:56,101 --> 00:02:00,235
...

38
00:02:00,259 --> 00:02:02,617
so signals act as a lighthouse

39
00:02:02,641 --> 00:02:07,807
... to locate, grow toward, reach

40
00:02:07,831 --> 00:02:11,037
and ... invade and colonize a plant.

41
00:02:11,427 --> 00:02:14,373

...identification of such signals

42
00:02:14,397 --> 00:02:19,022
... serves to elaborate strategy

43
00:02:19,046 --> 00:02:22,426
to block .... interaction between ... fungus and ... plant

44
00:02:22,752 --> 00:02:26,172
...lack ... appropriate method ...

45
00:02:26,196 --> 00:02:31,093
prevented ... identifying this mechanism at the molecular level

46
00:02:33,323 --> 00:02:36,450
...

47
00:02:36,474 --> 00:02:37,998
...

48
00:02:38,022 --> 00:02:41,355
...

49
00:02:41,379 --> 00:02:45,744
today ...

50
00:02:45,768 --> 00:02:50,592
... identify such plant signals

51
00:02:50,616 --> 00:02:53,973
by studying the interaction between a ... fungus

```
52
00:02:53,997 --> 00:02:55,680
...

53
00:02:55,704 --> 00:02:58,878
and one of its host plants...

54
00:03:00,310 --> 00:03:02,061
... characterize

55
00:03:02,085 --> 00:03:05,000
... receptor receiving ... signals

56
00:03:05,024 --> 00:03:08,720
and ... underlying reaction ... within the fungus

57
00:03:08,744 --> 00:03:11,963
and leading to ...growth toward ... plant.

58
00:03:12,879 --> 00:03:15,553
(Applause)

59
00:03:15,577 --> 00:03:16,728
Thank you.

60
00:03:16,752 --> 00:03:18,006
(Applause)

61
00:03:18,030 --> 00:03:20,793
... understanding of ... molecular processes
```

```
62
00:03:20,817 --> 00:03:23,315
... potential molecules

63
00:03:23,339 --> 00:03:27,139
... to create novel antifungal treatments

64
00:03:27,606 --> 00:03:30,002
.... treatments would disrupt

65
00:03:30,026 --> 00:03:32,765
... interaction between ... fungus and ... plant

66
00:03:32,789 --> 00:03:35,487
... blocking ... plant signal

67
00:03:35,511 --> 00:03:39,852
... the fungal reception system ...

68
00:03:39,876 --> 00:03:43,042
Fungal infections have devastated agriculture crops.

69
00:03:43,066 --> 00:03:45,788
...

70
00:03:45,812 --> 00:03:49,363
... demand of crop production ... increasing ...

71
00:03:49,387 --> 00:03:53,252
... due to population growth economic development,

72
```

```
00:03:53,276 --> 00:03:55,942
climate change ... demand for bio fuels.

73
00:03:56,751 --> 00:03:59,823
... understanding ...

74
00:03:59,847 --> 00:04:02,879
... interaction between ... fungus ... its host plant

75
00:04:02,903 --> 00:04:04,609
...

76
00:04:04,633 --> 00:04:09,974
... represents ... major step towards ... efficient strategy

77
00:04:09,998 --> 00:04:12,369
to combat plant fungal diseases

78
00:04:12,393 --> 00:04:15,918
....solving ... problems .... people's lives

79
00:04:15,942 --> 00:04:18,394
food security ... economic growth.

80
00:04:18,418 --> 00:04:19,570
Thank you.

81
00:04:19,594 --> 00:04:23,500
(Applause)
```

Listing 3: human-compressed subtitles.

```
1
```

```
00:00:00,825 --> 00:00:03,515
"Will the blight end the chestnut?

2
00:00:03,857 --> 00:00:05,857
The farmers rather guess not.

3
00:00:05,881 --> 00:00:08,119
It keeps smouldering at the roots

4
00:00:08,143 --> 00:00:10,151
And sending up new shoots

5
00:00:10,175 --> 00:00:11,841
Till another parasite

6
00:00:11,865 --> 00:00:14,269
Shall come to end the blight."

7
00:00:16,510 --> 00:00:18,549
...

8
00:00:18,573 --> 00:00:23,220
The American chestnut population,
counting nearly four billion trees

9
00:00:23,244 --> 00:00:26,345
was ... decimated by a fungal infection.

10
00:00:26,369 --> 00:00:29,577
Fungi are the most destructive pathogens of plants,

11
```

```
00:00:29,601 --> 00:00:32,323
...

12
00:00:32,673 --> 00:00:34,234
...

13
00:00:34,258 --> 00:00:37,125
Today crop losses associated with fungal infection

14
00:00:37,149 --> 00:00:41,065
are estimated at billions ...

15
00:00:41,585 --> 00:00:45,283
... represents ... food calories  ... half a billion people.

16
00:00:45,609 --> 00:00:47,967
And this leads to ... repercussions,

17
00:00:47,991 --> 00:00:51,411
including episodes of famine in developing countries

18
00:00:51,435 --> 00:00:54,839
large reduction of income for farmers and distributors

19
00:00:54,863 --> 00:00:56,791
high prices for consumers

20
00:00:56,815 --> 00:01:01,539
and risk of exposure to mycotoxin poison ...

21
00:01:02,318 --> 00:01:03,580
```

```
The ...

22
00:01:03,604 --> 00:01:06,321
is that the current method used to prevent and treat

23
00:01:06,345 --> 00:01:07,887
those dreadful diseases

24
00:01:07,911 --> 00:01:12,220
such as genetic control exploiting ... sources of resistance,

25
00:01:12,244 --> 00:01:15,625
... rotation ... treatment, ...

26
00:01:15,649 --> 00:01:18,331
are still limited ...

27
00:01:18,879 --> 00:01:21,299
They have to be constantly renewed.

28
00:01:21,323 --> 00:01:25,577
... we ... need to develop ... efficient strategies

29
00:01:25,601 --> 00:01:30,720
and for this research is required to identify biological mechanisms

30
00:01:30,744 --> 00:01:34,410
...

31
00:01:37,529 --> 00:01:40,663
... feature ... they cannot move
```

```
32
00:01:40,687 --> 00:01:44,212
and ... grow by extension...

33
00:01:44,236 --> 00:01:45,386
...

34
00:01:46,284 --> 00:01:50,537
... Anton de Bary, the ...

35
00:01:50,561 --> 00:01:54,117
was... presume ... fungi are guided by signals

36
00:01:54,141 --> 00:01:56,077
...plant,

37
00:01:56,101 --> 00:02:00,235
...plant ... it can ...

38
00:02:00,259 --> 00:02:02,617
... signals act as a lighthouse

39
00:02:02,641 --> 00:02:07,807
... fungi to locate, grow toward, reach

40
00:02:07,831 --> 00:02:11,037
and ... invade and colonize a plant.

41
00:02:11,427 --> 00:02:14,373
He knew that the identification of ... signals
```

42
00:02:14,397 --> 00:02:19,022
would unlock a great knowledge ... serves ...

43
00:02:19,046 --> 00:02:22,426
... block the interaction...

44
00:02:22,752 --> 00:02:26,172
...the lack of an appropriate method at that moment

45
00:02:26,196 --> 00:02:31,093
prevented him from identifying this mechanism at the molecular level

46
00:02:33,323 --> 00:02:36,450
Using purification and mutational genomic approaches

47
00:02:36,474 --> 00:02:37,998
.... technique

48
00:02:38,022 --> 00:02:41,355
allowing the measurement of directed ...

49
00:02:41,379 --> 00:02:45,744
today I'm glad ... 130 years,

50
00:02:45,768 --> 00:02:50,592
my former team and I could ... identify such plant signals

51
00:02:50,616 --> 00:02:53,973
by studying the interaction between a pathogenic fungus

52

00:02:53,997 --> 00:02:55,680
called Fusarium oxysporum

53
00:02:55,704 --> 00:02:58,878
and one of its host plants the tomato plant

54
00:03:00,310 --> 00:03:02,061
...we could characterize

55
00:03:02,085 --> 00:03:05,000
the fungal receptor receiving those signals

56
00:03:05,024 --> 00:03:08,720
... part of the ... reaction occurring ...

57
00:03:08,744 --> 00:03:11,963
and leading to its direct growth toward the plant

58
00:03:12,879 --> 00:03:15,553
(Applause)

59
00:03:15,577 --> 00:03:16,728
Thank you.

60
00:03:16,752 --> 00:03:18,006
(Applause)

61
00:03:18,030 --> 00:03:20,793
The understanding ...of ... molecular processes

62
00:03:20,817 --> 00:03:23,315

offers a panel of potential molecules

63
00:03:23,339 --> 00:03:27,139
...

64
00:03:27,606 --> 00:03:30,002
And those treatments would disrupt

65
00:03:30,026 --> 00:03:32,765
the interaction between the fungus and the plant

66
00:03:32,789 --> 00:03:35,487
...

67
00:03:35,511 --> 00:03:39,852
...

68
00:03:39,876 --> 00:03:43,042
Fungal infections have devastated agriculture crops.

69
00:03:43,066 --> 00:03:45,788
Moreover, we are now in an era

70
00:03:45,812 --> 00:03:49,363
where the demand of crop production is increasing ...

71
00:03:49,387 --> 00:03:53,252
... is due ...

72
00:03:53,276 --> 00:03:55,942
climate change and demand for bio fuels

```
73
00:03:56,751 --> 00:03:59,823
Our understanding of the molecular mechanism

74
00:03:59,847 --> 00:04:02,879
of interaction between a fungus and its host plant,

75
00:04:02,903 --> 00:04:04,609
such ... plant,

76
00:04:04,633 --> 00:04:09,974
potentially represents a ... towards developing ... efficient strategy

77
00:04:09,998 --> 00:04:12,369
... to ...

78
00:04:12,393 --> 00:04:15,918
...solving...

79
00:04:15,942 --> 00:04:18,394
...

80
00:04:18,418 --> 00:04:19,570
Thank you.

81
00:04:19,594 --> 00:04:23,500
(Applause)
```

Listing 4: system-compressed subtitles

A.3 Example Questionnaire

Subtitles 1

As part of my Master Thesis I investigate the effect of simplified subtitles on spoken language understanding.

In this questionnaire, you will see nine videos with 3 comprehension questions each, also you will be asked to give subjective feedback after each video.
The study will take approximately 60 minutes.

Please watch the videos completely and only once. Answer the questions completely and meaningfully.

Rest assured that this study doesn't want to test you, but rather the concept of simplified subtitles.

Your data is stored anonymously and will only be used for research purposes. After the evaluation of this study it will be deleted.

Thank you for taking the time to do my user study :)

Katrin Angerbauer (experimenter)

Dr. Heike Adel and Prof. Dr. Ngoc Thang Vu (supervisors)

There are 64 questions in this survey.

Please insert your participant id that you were given at the beginning of the study. *

Please write your answer here:

What is your gender? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ female
- ◯ male
- ◯ other
- ◯ prefer not to answer
- ◯ prefer to self-describe: _____

What is your age? *

❗ Only an integer value may be entered in this field.
Please write your answer here:

_____

Please specify your occupation *

Please write your answer here:

_____

What is your mother tongue? *

Choose one of the following answers
Please choose **only one** of the following:

- ◯ German
- ◯ English
- ◯ Other

Please state your proficiency level of the English language *

Choose one of the following answers
Please choose **only one** of the following:

- ◯ native speaker
- ◯ fluent (C1-C2)
- ◯ good knowlegde (B1-B2)
- ◯ basic skills (A1-A2)

Please choose one option for each statement. *

Please choose the appropriate response for each item:

	Never	Rarely	Sometimes	Often	Always
How frequently do you listen to English content (films, podcasts, audiobooks, lectures ....)	◯	◯	◯	◯	◯
How often do you use subtitles when watching videos in English	◯	◯	◯	◯	◯

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

According to the speaker what are we taking for granted? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ Access to information
- ◯ Access to computers
- ◯ Access to public libraries
- ◯ I don’t know.

What is the memex by Vannevar Bush? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ I don’t know
- ◯ Personal library of the articles and books one has access to.
- ◯ One of the first internet browsers.
- ◯ An editor to capture new ideas.

To what thing is the hyperlink compared to? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ A LEGO Block
- ◯ A thread
- ◯ A brick
- ◯ I don't know

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

What **CANNOT** be determined by Kepler's observation *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ Size of the planet
- ◯ UV rays and X rays a planet receives
- ◯ Distance to the parent star
- ◯ I don't know.

What are the so-called sunspots evidence for? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ The sun's electric field
- ◯ The sun's radiation
- ◯ The sun's magnetic field
- ◯ I don't know.

According to the speaker what sets the stage for life in the universe? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ Water
- ◯ Starlight
- ◯ Comets
- ◯ I don't know.

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

Why does the speaker think the pencil is perfect? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ It can be erased.
◯ It has a long history of collaboration
◯ It is a simple object.
◯ I don't know.

What material is the core of the pencil NOT made of? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ Coal
◯ Clay
◯ Water
◯ I don't know

What is responsible for the hardness of the pencil? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ Graphite
◯ Clay
◯ Wood
◯ I don't know.

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

What is NOT said about Arabic? *

Choose one of the following answers
Please choose **only one** of the following:

- ◯ It is similar to Persian
- ◯ It is logical
- ◯ It is difficult to pronounce for Europeans
- ◯ I don't know

How did mathematics come to Europe? *

Choose one of the following answers
Please choose **only one** of the following:

- ◯ Via Spain
- ◯ Via Portugal
- ◯ Via Gibraltar
- ◯ I don’t know.

What sound is difficult in Spanish? *

Choose one of the following answers
Please choose **only one** of the following:

- ◯ The CK sound
- ◯ The SH sound
- ◯ The CH sound
- ◯ I don’t know

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

What is the name of the Saturn moon the speaker talks about? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ Phoebe
◯ Cassini
◯ Enceladus
◯ I don't know.

What is NOT mentioned as part of the organic compounds? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ Cyanide
◯ Formaldehyde
◯ Oxygen
◯ I don’t know.

What is a circumstance that could sustain life? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ Liquid water in contact with rocks.
◯ Ice in contact with rocks.
◯ Carbon dioxide in contact with rocks.
◯ I don’t know.

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

Where are the earliest hoodies from? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ Ancient Greece and ancient Rome
◯ Ancient China and the Orient
◯ Ancient Rome and the Orient
◯ I don’t know.

What physiological or psychological element of wearing a hoodie is NOT mentioned in the talk? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ It keeps you warm.
◯ It makes you feel protected.
◯ It makes you feel invisible.
◯ I don’t know.

Who shot Trayvon Martin? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ a policeman
◯ a vigilante
◯ a gang member
◯ I don't know.

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

What assumption did Anton de Bary make in 1884? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ Fungi are guided by signals from the host plant
- ◯ Fungi are guided by signals from other fungi
- ◯ Fungi are mislead by signals from the host plant.
- ◯ I don't know

Which interaction did the talker and her team study? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ Fungus Fusarium oxysporum and the tomato plant.
- ◯ Fungus Fusarium oxysporum and the pepper plant.
- ◯ Fungus Fusarium oxysporum and barley.
- ◯ I don’t know..

What reason is NOT mentioned in regard to the increasing demand of crop production? *

❗ Choose one of the following answers
Please choose **only one** of the following:

- ◯ Population growth
- ◯ Climate change
- ◯ Hunger in the world
- ◯ I don’t know.

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

What was the purpose of the ledger according to the speaker? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ To show how much of a task was completed by the computer.
◯ To show how much of a task was completed at a factory.
◯ For people to fill in their working hours.
◯ I don't know.

What caused the "software crisis"? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ Computers were low on demand.
◯ Computers were not fast enough.
◯ Computers were getting complicated.
◯ I don't know.

What did Brad Myers want to study? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ The effect of the design of a progress bar.
◯ The effect on user experience.
◯ The causes of long processing times.
◯ I don't know.

Kein Video mit unterstütztem Format und MIME-Typ gefunden.

Please select an option for each statement *

Please choose the appropriate response for each item:

	Very Low	Low	Moderate	High	Very High
Mental demand: How mentally demanding was it to read the subtitles?	◯	◯	◯	◯	◯
Temporal demand: How rushed did you feel when reading the subtitles?	◯	◯	◯	◯	◯
Effort: How hard did you have to concentrate to follow the subtitles as well as the video to understand what's going on?	◯	◯	◯	◯	◯
Frustation: How irritated by the subtitles were you when watching the video?	◯	◯	◯	◯	◯

Please choose one of the following options. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I was familiar with the topic of the video.	◯	◯	◯	◯	◯
The subtitles were easy to read.	◯	◯	◯	◯	◯
The subtitles helped me to understand the content.	◯	◯	◯	◯	◯
The subtitles were confusing.	◯	◯	◯	◯	◯
The subtitles were too short.	◯	◯	◯	◯	◯
The subtitles were too long.	◯	◯	◯	◯	◯
The subtitles contained all important information.	◯	◯	◯	◯	◯

What is NOT TRUE about the Hoag's Object? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ It has an outer ring.
◯ It is rare.
◯ It is spiral.
◯ I don't know.

What is special about the newly discovered galaxy? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ It is an Hoag's Object.
◯ It is very far from earth.
◯ It has an inner ring.
◯ I don't know.

What does the speaker hope to gain by studying this rare galaxy? *

❗ Choose one of the following answers
Please choose **only one** of the following:

◯ New clues on how the universe works.
◯ Insights about our solar system.
◯ New theories on black holes.
◯ I don't know.

Please rate your subtitle experience. *

Please choose the appropriate response for each item:

	Disagree	Somewhat disagree	Neither agree or disagree	Somewhat agree	Agree
I liked the subtitles with fewer words	◯	◯	◯	◯	◯
I liked the subtitles with no words left out	◯	◯	◯	◯	◯

I would like to have shortened subtitles for the following contexts *

Check all that apply

Please choose **all** that apply:

- ☐ Lectures
- ☐ Movies
- ☐ Talks
- ☐ Other:

Other things you like to mention:

Please write your answer here:

Thank you :) You can collect your reward from the experimenter ;)

Submit your survey.
Thank you for completing this survey.

B Further Experiment Results

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_plain	120	no_punct	0.186	0.841	0.299	0.790	0.900
Simple-LSTM_plain	120	punct*	0.191	0.850	0.330	0.791	0.914
Simple-LSTM_plain	120	selected_punct *	0.199	0.850	0.330	0.790	0.922
Simple-LSTM_plain	256	no_punct	0.197	0.845	0.305	0.796	0.902
Simple-LSTM_plain	256	all_punct*	0.200	0.854	0.320	0.797	0.915
Simple-LSTM_plain	256	selected_punct *	0.202	0.854	0.322	0.797	0.912

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_POS (no_punct, 120)	120	no_punct	0.210	0.852	0.308	0.804	0.912
Simple-LSTM_POS	120	all_punct*	0.217	0.859	0.327	0.808	0.909
Simple-LSTM_POS	120	selected_punct *	0.214	0.857	0.328	0.806	0.910
Simple-LSTM_POS	256	no_punct	0.212	0.853	0.307	0.806	0.909
Simple-LSTM_POS	256	all_punct*	0.221	0.859	0.330	0.804	0.923
Simple-LSTM_POS	256	selected_punct *	0.224	0.859	0.330	0.806	0.920

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_previous	120	no_punct	0.194	0.84	0.300	0.789	0.900
Simple-LSTM_previous	120	all_punct*	0.200	0.852	0.329	0.793	0.919
Simple-LSTM_previous	120	selected_punct *	0.194	0.849	0.318	0.791	0.914
Simple-LSTM_previous	256	no_punct	0.195	0.842	0.300	0.79	0.909
Simple-LSTM_previous	256	all_punct*	0.200	0.852	0.330	0.795	0.917
Simple-LSTM_previous	256	selected_punct *	0.201	0.851	0.326	0.794	0.917

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_compression	120	no_punct	0.231	0.866	0.405	0.829	0.902
Simple-LSTM_compression	120	all_punct*	0.231	0.87	0.379	0.886	0.832
Simple-LSTM_compression	120	selected_punct *	0.238	0.866	0.410	0.828	0.905
Simple-LSTM_compression	256	no_punct	0.234	0.870	0.379	0.833	0.887
Simple-LSTM_compression	256	all_punct*	0.242	0.869	0.360	0.834	0.880
Simple-LSTM_compression	256	selected_punct *	0.233	0.866	0.374	0.827	0.887

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_POS_previous	120	no_punct	0.220	0.852	0.310	0.804	0.912
Simple-LSTM_POS_previous	120	all_punct*	0.214	0.855	0.325	0.795	0.928
Simple-LSTM_POS_previous	120	selected_punct *	0.222	0.856	0.330	0.800	0.923
Simple-LSTM_POS_previous	256	no_punct	0.215	0.853	0.308	0.806	0.910
Simple-LSTM_POS_previous	256	all_punct*	0.231	0.862	0.332	0.809	0.920
Simple-LSTM_POS_previous	256	selected_punct *	0.233	0.859	0.326	0.804	0.926

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_POS_compression	120	no_punct	0.259	0.873	0.399	0.839	0.905
Simple-LSTM_POS_compression	120	all_punct*	0.259	0.876	0.376	0.840	0.889
Simple-LSTM_POS_compression	120	selected_punct *	0.254	0.875	0.384	0.840	0.893
Simple-LSTM_POS_compression	256	no_punct	0.260	0.875	0.419	0.840	0.910
Simple-LSTM_POS_compression	256	all_punct*	0.263	0.876	0.383	0.840	0.891
Simple-LSTM_POS_compression	256	selected_punct *	0.256	0.878	0.387	0.842	0.896

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_previous_compression	120	no_punct	0.243	0.864	0.390	0.825	0.900
Simple-LSTM_previous_compression	120	all_punct*	0.239	0.872	0.374	0.835	0.884
Simple-LSTM_previous_compression	120	selected_punct *	0.241	0.866	0.383	0.826	0.889
Simple-LSTM_previous_compression	256	no_punct	0.248	0.866	0.398	0.829	0.900
Simple-LSTM_previous_compression	256	all_punct*	0.242	0.874	0.387	0.836	0.891
Simple-LSTM_previous_compression	256	selected_punct *	0.241	0.867	0.369	0.83	0.880

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Simple-LSTM_POS_previous_compression	120	no_punct	0.232	0.863	0.348	0.829	0.885
Simple-LSTM_POS_previous_compression	120	all_punct*	0.266	0.879	0.379	0.845	0.891
Simple-LSTM_POS_previous_compression	120	selected_punct *	0.268	0.878	0.380	0.843	0.895
Simple-LSTM_POS_previous_compression	256	no_punct	0.266	0.872	0.419	0.834	0.918
Simple-LSTM_POS_previous_compression	256	all_punct*	0.268	0.879	0.397	0.843	0.899
Simple-LSTM_POS_previous_compression	256	selected_punct *	0.275	0.877	0.376	0.842	0.895

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Seq2Seq-LSTM_previous_compression	120	no_punct	0.287	0.876	0.389	0.843	0.900
Seq2Seq-LSTM_previous_compression	120	all_punct*	0.258	0.879	0.361	0.845	0.886
Seq2Seq-LSTM_previous_compression	120	selected_punct *	0.282	0.883	0.399	0.849	0.900
Seq2Seq-LSTM_previous_compression	256	no_punct	0.301	0.876	0.411	0.843	0.904
Seq2Seq-LSTM_previous_compression	256	all_punct*	0.270	0.880	0.371	0.847	0.889
Seq2Seq-LSTM_previous_compression	256	selected_punct *	0.291	0.884	0.393	0.851	0.896
Seq2Seq-LSTM_previous_compression							

model	hidden dimensionality	punctuation evaluation	Accuracy sentences	Accuracy tokens	Accuracy compression ratio	f1_KEEP	f1_DELETE
Seq2Seq-LSTM_POS_previous_compression	120	no_punct	0.318	0.885	0.421	0.854	0.912
Seq2Seq-LSTM_POS_previous_compression	120	all_punct*	0.293	0.889	0.381	0.857	0.898
Seq2Seq-LSTM_POS_previous_compression	120	selected_punct *	0.294	0.883	0.366	0.853	0.887
Seq2Seq-LSTM_POS_previous_compression	256	no_punct	0.327	0.888	0.410	0.859	0.911
Seq2Seq-LSTM_POS_previous_compression	256	all_punct*	0.316	0.892	0.417	0.859	0.909
Seq2Seq-LSTM_POS_previous_compression	256	selected_punct *	0.315	0.890	0.383	0.859	0.897

Bibliography

Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, and Michael Collins. Globally Normalized Transition-Based Neural Networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2442–2452, Stroudsburg, PA, USA, 2016. Association for Computational Linguistics.

Alan Baddeley. Working memory. *Science*, 255(5044):556–559, 1992.

Deni Basaraba, Paul Yovanoff, Julie Alonzo, and Gerald Tindal. Examining the structure of reading comprehension: do literal, inferential, and evaluative comprehension truly exist? *Reading and Writing*, 26(3):349–379, 2013.

Sorayya Behroozizad and Sudabeh Majidi. The Effect of Different Modes of English Captioning on EFL Learners' General Listening Comprehension: Full Text vs. Keyword Captions. *Advances in Language and Literary Studies*, 6(4):115–121, 2015.

Elias Adam Bensalem. The impact of keyword and full video captioning on listening comprehension. *Journal of Teaching English for Specific and Academic Purposes*, 4(3):453 – 463, 2016.

Larwan Berke, Christopher Caulfield, and Matt Huenerfauth. Deaf and Hard-of-Hearing Perspectives on Imperfect Automatic Speech Recognition for Captioning One-on-One Meetings. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '17*, pages 155–164, New York, New York, USA, 2017. ACM Press.

Andy Brown, Rhia Jones, Mike Crabb, James Sandford, Matthew Brooks, Mike Armstrong, and Caroline Jay. Dynamic Subtitles: The User Experience. In *Proceedings of the ACM International Conference on Interactive*

Experiences for TV and Online Video (TVX), pages 103–112, New York, New York, USA, 2015. ACM Press.

Denis Burnham, Greg Leigh, William Noble, Caroline Jones, Michael Tyler, Leonid Grebennikov, and Alex Varley. Parameters in Television Captioning for Deaf and Hard-of-Hearing Adults: Effects of Caption Rate Versus Text Reduction on Comprehension. *Journal of Deaf Studies and Deaf Education*, 13(3):391–404, 2008.

Raman Chandrasekar, Christine Doran, and Srinivas Bangalore. Motivations and methods for text simplification. In *Proceedings of the 16th Conference on Computational Linguistics (COLING)*, volume 2, pages 1041–1044, Morristown, NJ, USA, 1996. Association for Computational Linguistics.

Edwin Chen. Exploring LSTMs, 2018. URL `http://blog.echen.me/2017/05/30/exploring-lstms/`.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNL)*, pages 1724–1734. Association for Computational Linguistic, 2014.

James Clarke and Mirella Lapata. Models for Sentence Compression: A Comparison across Domains, Training Requirements and Evaluation Measures. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, pages 377–384. Association for Computational Linguistics, 2006.

James Clarke and Mirella Lapata. Global Inference for Sentence Compression: An Integer Linear Programming Approach. *Journal of Artificial Intelligence Research*, 31:399–429, mar 2008.

Trevor Cohn and Mirella Lapata. Large Margin Synchronous Generation and its Application to Sentence Compression. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 73–82. Association for Computational Linguistics, 2007.

Trevor Cohn and Mirella Lapata. Sentence Compression Beyond Word Deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, pages 137–144. Association for Computational Linguistics, 2008.

Trevor Cohn and Mirella Lapata. Sentence Compression as Tree Transduction. *Journal of Artificial Intelligence Research*, 34:637–674, 2009.

Yue Dong. A Survey on Neural Network-Based Summarization Methods. *Computing Research Repository (CoRR)*, mar 2018.

Radim Řehůřek and Petr Sojka. Software framework for topic modelling with large corpora. In *Proceedings of LREC 2010 Workshop New Challenges for NLP Frameworks*, pages 45–50. European Language Resources Association (ELRA), 2010.

European Council. The CEFR Levels. URL `https://www.coe.int/en/web/common-european-framework-reference-languages/level-descriptions`.

Veri Ferdiansyah and Seiichi Nakagawa. Effect of captioning lecture videos for learning in foreign language. Technical Report 13, Toyohashi University of Technology, 2013.

Andy P. Field and Graham J. Hole. Descriptive Statistics. In *How to Design and Report Experiments*, pages 109–140. Sage Publications, London, England, 2003.

Katja Filippova and Yasemin Altun. Overcoming the Lack of Parallel Data in Sentence Compression. In *Proceedings of the 2013 Conference on Empirical*

Methods in Natural Language Processing, pages 1481–1491. Association for Computational Linguistics, 2013.

Katja Filippova and Michael Strube. Dependency tree based sentence compression. In *Proceedings of the 5th International Natural Language Generation Conference*, pages 25–32. Association for Computational Linguistics, 2008.

Katja Filippova, Enrique Alfonseca, Carlos A. Colmenares, Lukasz Kaiser, and Oriol Vinyals. Sentence Compression by Deletion with LSTMs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 360–368, Stroudsburg, PA, USA, 2015. Association for Computational Linguistics.

Goran Glavaš and Sanja Štajner. Simplifying lexical simplification: do we need simplified corpora? In *53rd annual meeting of the Association for Computational Linguistics and 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 63–68. Association for Computational Linguistics, 2015.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 249–256. PMLR, 2010.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. URL `http://www.deeplearningbook.org/`.

Tom Grimes. Mild Auditory-Visual Dissonance in Television News May Exceed Viewer Attentional Capacity. *Human Communication Research*, 18 (2):268–298, 1991.

Helen Gant Guillory. The Effects of Keyword Captions to Authentic French Video on Learner Comprehension. *CALICO Journal*, 15(1-3):89–108, 1998.

Han Guo, Ramakanth Pasunuru, and Mohit Bansal. Dynamic Multi-Level Multi-Task Learning for Sentence Simplification. In Emily M. Bender, Leon Derczynski, and Pierre Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 426–476, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics.

Sandra G. Hart and Lowell E. Staveland. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology*, 52(C):139–183, jan 1988.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 1997.

Kentaro Inui, Atsushi Fujita, Tetsuro Takahashi, Ryu Iida, and Tomoya Iwakura. Text Simplification for Reading Assistance. In *Proceedings of the 2nd International Workshop on Paraphrasing (PARAPHRASE)*, volume 16, pages 9–16, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

Hongyan Jing and Hongyan. Sentence reduction for automatic text summarization. In *Proceedings of the 6th Conference on Applied Natural Language Processing*, pages 310–315, Morristown, NJ, USA, 2000. Association for Computational Linguistics.

Karen Sparck Jones. Automatic summarising: factors and directions. In Mark T. Maybury and Inderjeet Mani, editors, *Advances in Automatic Text Summarization*, pages 1–12, Cambridge, MA, USA, 1999. MIT Press.

Mikael Kågebäck, Olof Mogren, Nina Tahmasebi, and Devdatt Dubhashi. Extractive Summarization using Continuous Vector Space Models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) EACL 2014*, pages 31–39. Association for Computational Linguistics, 2014.

Fotios Karamitroglou. A Proposed Set of Subtitling Standards in Europe. *Translation Journal [Web Version]*, 2(2), 1998. URL `https://translationjournal.net/journal/04stndrd.htm`.

Saba Kawas, George Karalis, Tzu Wen, and Richard E. Ladner. Improving Real-Time Captioning Experiences for Deaf and Hard of Hearing Students. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*, pages 15–23, New York, New York, USA, 2016. ACM Press.

Nikhil Ketkar. Introduction to PyTorch. In *Deep Learning with Python*, pages 195–208. Apress, Berkeley, CA, 2017.

Jane King. Using DVD Feature Films in the EFL Classroom. *Computer Assisted Language Learning*, 15(5):509–523, 2002.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2014.

Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. Improving sentence compression by learning to predict gaze. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1528–1533, 2016.

Kevin Knight and Daniel Marcu. Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, 139(1):91–107, 2002.

Cees M. Koolstra, Allerd L. Peeters, and Herman Spinhof. The Pros and Cons of Dubbing and Subtitling. *European Journal of Communication*, 17 (3):325–354, 2002.

Izabela Krejtz, Agnieszka Szarkowska, and Maria Łogińska. Reading Function and Content Words in Subtitled Videos. *Journal of Deaf Studies and Deaf Education*, 21(2):222–232, 2016.

Jan-Louis Kruger, Esté Hefer, and Gordon Matthew. Measuring the impact of subtitles on cognitive load. In *Proceedings of the 2013 Conference on Eye Tracking South Africa - ETSA '13*, pages 62–66. ACM Press, 2013.

Kuno Kurzhals, Emine Cetinkaya, Yongtao Hu, Wenping Wang, and Daniel Weiskopf. Close to the Action. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*, pages 6559–6568, New York, New York, USA, 2017. ACM Press.

Dac-Viet Lai, Nguyen Truong Son, and Nguyen Le Minh. Deletion-Based Sentence Compression Using Bi-enc-dec LSTM. In *International Conference of the Pacific Association for Computational Linguistics*, pages 249–260. Springer, Singapore, 2017.

Zhonglei Lu, Wenfen Liu, Yanfang Zhou, Xuexian Hu, and Binyu Wang. An Effective Approach of Sentence Compression Based on "Re-read" Mechanism and Bayesian Combination Model. In *Chinese National Conference on Social Media Processing*, pages 129–140. Springer, Singapore, 2017.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. Evaluation in information retrieval. In *Introduction to Information Retrieval*, chapter 8, pages 154–175. Cambridge University Press, 2009.

Paul L. Markham. Captioned videotapes and second-language listening word recognition. *Foreign Language Annals*, 32(3):321–328, oct 1999.

Anna Matamala, Pilar Orero, Sara Rovira-Esteva, Helena Casas Tost, Fernando Morales Morante, Olga Soler Vilageliu, Belén Agulló, Anita Fidyka, Daniel Segura Giménez, and Irene Tor-Carroggio. User-centric approaches in access services evaluation: profiling the end user. In *Proceedings of the Eleventh International Conference on Language Resources Evaluation (LREC 2018)*, pages 1–7. European Language Resources Association (ELRA), 2018.

Richard E. Mayer and Roxana Moreno. Aids to computer-based multimedia learning. *Learning and Instruction*, 12(1):107–119, 2002. ISSN 0959-4752. doi: 10.1016/S0959-4752(01)00018-4.

Richard E. Mayer and Roxana Moreno. Nine Ways to Reduce Cognitive Load in Multimedia Learning. *Educational Psychologist*, 38(1):43–52, mar 2003. ISSN 0046-1520. doi: 10.1207/S15326985EP3801_6.

Matthias R Mehl, Simine Vazire, Nairán Ramírez-Esparza, Richard B Slatcher, and James W Pennebaker. Are women really more talkative than men? *Science*, 317(5834):82, jul 2007.

Maryam Sadat Mirzaei, Kourosh Meshgi, Yuya Akita, and Tatsuya Kawahara. Partial and synchronized captioning: A new tool to assist learners in developing second language listening skill. *ReCALL - The Journal of the European Association for Computer Assisted Language Learning*, 29 (2):178–199, 2017.

Mohammad Reza Mirzaei, Seyed Ghorshi, and Mohammad Mortazavi. Audio-visual speech recognition techniques in augmented reality environments. *The Visual Computer*, 30(3):245–257, 2014.

Maribel Montero Perez, Elke Peters, and Piet Desmet. Is less more? Effectiveness and perceived usefulness of keyword and full captioned video for L2 listening comprehension. *ReCALL - The Journal of the European Association for Computer Assisted Language Learning*, 26(1):21–43, 2014.

Sioban Moran. The effect of linguistic variation on subtitle reception. In Elisa Perego, editor, *Eyetracking in Audiovisual Translation*, pages 183–222. Aracne Editrice, 2012.

Ramesh Nallapati, Bowen Zhou, Cicero Nogueira dos Santos, Caglar Gulcehre, and Bing Xiang. Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond. In *Proceedings of the 20th SIGNLL Con-*

ference on Computational Natural Language Learning (CoNLL), pages 280–290. Association for Computational Linguistic, 2016.

Ani Nenkova and Kathleen McKeown. A survey of text summarization techniques. In Charu C. Aggarwal and Cheng Xiang Zhai, editors, *Mining Text Data*, chapter 3, pages 43–76. Springer US, Boston, MA, 2012.

Michael A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2015. URL `http://neuralnetworksanddeeplearning.com/`.

Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P. Dinu. Exploring Neural Text Simplification Models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 85–91, Stroudsburg, PA, USA, 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-2014.

Christopher Olah. Understanding LSTM Networks, 2015. URL `http://colah.github.io/posts/2015-08-Understanding-LSTMs/`.

Sarah E Petersen and Mari Ostendorf. Text Simplification for Language Learners: A Corpus Analysis. In *Proceedings of the Workshop on Speech and Language Technology in Education (SLaTE)*, pages 69–72. Carnegie Mellon University and ISCA Archive, 2007.

Agnès Piquard-Kipffer, Odile Mella, Jérémy Miranda, Denis Jouvet, and Luiza Orosanu. Qualitative investigation of the display of speech recognition results for communication with deaf people. In *6th Workshop on Speech and Language Processing for Assistive Technologies*, page 7. ACL/ISCA, 2015.

Raisa Rashid, Quoc Vy, Richard G. Hunt, and Deborah I. Fels. Dancing with words. In *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition - C&C '07*, page 269, New York, New York, USA, 2007. ACM Press.

Tariq Rashid. *Make your own neural network (German Version)*. CreateSpace Independent Publishing Platform, 1 edition, 2016.

Keith Rayner. Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, 124(3):372–422, 1998.

Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Jun Ma, and Maarten de Rijke. Leveraging Contextual Sentence Relations for Extractive Summarization Using a Neural Attention Model. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 95–104, New York, New York, USA, 2017. ACM Press.

Kevin Rooney. The Impact of Keyword Caption Ratio on Foreign Language Listening Comprehension. *International Journal of Computer-Assisted Language Learning and Teaching*, 4(2):11–28, 2014.

Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

Alexander M. Rush, Sumit Chopra, and Jason Weston. A Neural Attention Model for Abstractive Sentence Summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389. Association for Computational Linguistics, 2015.

Rufat Rzayev, Paweł W. Wozniak, Tilman Dingler, and Niels Henze. Reading on Smart Glasses. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI)*, pages 1–9, New York, New York, USA, 2018. ACM Press.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get To The Point: Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1073–1083. Association for Computational Linguistics, 2017.

Matthew Shardlow. A Survey of Automated Text Simplification. *International Journal of Advanced Computer Science and Applications*, 4(1): 58–70, 2014.

Advaith Siddharthan. A survey of research on text simplification. *International Journal of Applied Linguistics (ITL)*, 165(2):259–298, 2015.

Lucia Specia. Translating from complex to simplified sentences. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6001 LNAI:30–39, 2010.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to Sequence Learning with Neural Networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS)*, pages 3104–3112, Cambridge, MA, USA, 2014. MIT Press.

Nhi-Thao Tran, Viet-Thang Luong, Ngan Luu-Thuy Nguyen, and Minh-Quoc Nghiem. Effective attention-based neural architectures for sentence compression with bidirectional long short-term memory. In *Proceedings of the Seventh Symposium on Information and Communication Technology - SoICT '16*, pages 123–130, New York, New York, USA, 2016. ACM Press.

Wade M Vagias. Likert-type Scale Response Anchors. Clemson International Institute for Tourism. *& Research Development, Department of Parks, Recreation and Tourism Management, Clemson University*, 2006.

Robert Vanderplank. The value of teletext sub-titles in language learning. *ELT Journal*, 42(4):272–281, 1988.

Mike Wald. Creating accessible educational multimedia through editing automatic speech recognition captioning in real time. *Interactive Technology and Smart Education*, 3(2):131–141, 2006.

Liangguo Wang, Jing Jiang, Hai Leong Chieu, Chen Hui Ong, Dandan Song, and Lejian Liao. Can Syntax Help? Improving an LSTM-based Sentence

Compression Model for New Domains. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1385–1393. Association for Computational Linguistic, 2017.

Phillip Ward, Ye. Wang, Peter. Paul, and Mardi. Loeterman. Near-Verbatim Captioning Versus Edited Captioning for Students Who Are Deaf or Hard of Hearing: A Preliminary Investigation of Effects on Comprehension. *American Annals of the Deaf*, 152(1):20–28, 2007.

Helen Williams and David Thorne. The value of teletext subtitling as a medium for language learning. *System*, 28(2):217–228, 2000.

Paula Winke, Susan Gass, and Tetyana Sydorenko. The effects of captioning videos used for foreign language listening activities. *Language Learning & Technology*, 14(1):66–87, 2010.

Sander Wubben, Antal van den Bosch, and Emiel Krahmer. Sentence simplification by monolingual machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, pages 1015–1024. Association for Computational Linguistics, 2012.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. Optimizing Statistical Machine Translation for Text Simplification. *Transactions of the Association for Computational Linguistics (TACL)*, 4: 401–415, 2016.

Jie Chi Yang, Chia Ling Chang, Yi Lung Lin, and Mei Jen Audrey Shih. A study of the POS keyword caption effect on listening comprehension. In *Proceedings of the 18th International Conference on Computers in Education: Enhancing and Sustaining New Knowledge Through the Use of Digital Technology in Education, ICCE 2010*, pages 708–712. ACM Press, 2010.

Noa Talaván Zanón. Using subtitles to enhance foreign language learning.

Porta Linguarum: revista internacional de didáctica de las lenguas extranjeras, 6:4, 2006.

Justin Jian Zhang, Ricky Ho Yin Chan, and Pascale Fung. Extractive speech summarization using shallow rhetorical structure modeling. *IEEE Transactions on Audio, Speech and Language Processing*, 18(6):1147–1157, 2010.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1353–1361. Association for Computational Linguistics, 2010.

All links were last followed on 13.12.2018.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

---

place, date, signature