FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Dávid Javorský

Presentation of simultaneous speech translation

Institute of Formal and Applied Linguistics

Supervisor of the master thesis: doc. RNDr. Ondřej Bojar, Ph.D.
Study programme: Computer Science
Study branch: Computational Linguistics

Prague 2021

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date ....

Author's signature

I would like to thank my supervisor, doc. RNDr. Ondřej Bojar, Ph.D., for his enthusiasm for the project, patient assistance and expert guidance through each stage of the process.

I cannot forget to thank my family and friends for their invaluable support during this very intense academic year.

Title: Presentation of simultaneous speech translation

Author: Dávid Javorský

Institute: Institute of Formal and Applied Linguistics

Supervisor: doc. RNDr. Ondřej Bojar, Ph.D., Institute of Formal and Applied Linguistics

Abstract: The goal of the thesis is to examine methods for presenting the outputs of simultaneous speech translation systems to users. Compared to manually edited subtitles in movies, current translation systems achieve lower quality in automatic subtitling due to limitations of reading speed and space constraints. Therefore, this thesis examines possibilities of relaxing these constraints, achieving an improvement in overall quality of translated speech presentation. The work relies on existing translation and speech recognition systems, proposing methods solely for user evaluation of the presentation. It provides a thorough analysis of selecting the best layout from a number of presentation options, recommending its usage given available resources. The experimenting is performed given our proposed interface for subtitling and testing. Additionally, the results bring new insights to the relation between the evaluation methods, suggesting less time-consuming evaluation for specific usecases in future experiments.

Keywords: simultaneous speech translation, automatic subtitling, user evaluation

Contents

Introduction

Machine translation (MT) is nowadays commonly used in many areas of language processing. Automatic speech recognition (ASR) extends possibilities for MT usage, e.g. in automatic simultaneous speech translation. ASR usually returns unstable very last output which can be required to be shown in a short period of time and limited space. In practice, this involves displaying subtitles on a video or providing one screen with subtitles for several translated languages.

Research of language and cultural rules led to broad recommendations for proper subtitling [Karamitroglou, 1998, Williams, 2009]. Attempts to automatically provide subtitles given translated simultaneous speech recognition resulted to a pipeline of multiple-level ASR, postprocessed MT and a "subtitler" implemented in Perl [Macháček and Bojar, 2020]. This so-called "subtitler" is a component that stabilizes the flow of text to make it easier to follow for the user given a fixed small area of the screen. Further work brought a lightweight version of the Perl subtitler reimplemented for a web browser [Smrž, 2020], utilizing the same communication protocol. However, a study of balancing parameters for effective subtitled presentation is missing yet.

The goal of this thesis is to empirically investigate viewer preference of preconfigured so-called *subtitling layouts*. The layouts are defined as a combination of spatial, positional, typographical and visual characteristics of presented subtitles. The preference of viewers is assumed to be derived from the amount of caught information from presentation, shortly denoting as their *comprehension*. Although comprehension heavily depends on the quality of used translation systems, evaluating these systems separately is not sufficient to obtain the whole picture of viewer's experience.

Our main evaluation methods are (1) derived from the previous work which utilizes *continuous rating* as immediate feedback of user satisfaction with the presented subtitles [Macháček and Bojar, 2020], and (2) *questionnaires* which are partially inspired from manual evaluation of MT output [Berka et al., 2011]. Besides the main goal of this thesis — to experimentally analyze user comprehension when employing different layouts — more thorough analysis reveals additional findings about other presentation properties (e.g latency or fluency), relation between evaluation methods and the source of caught information.

The experiments are conducted using our proposed implementation of the subtitler with a customizable user configuration. We also design and implement an interface for experiment management, maintenance and distribution.

We emphasize that this work is a pilot study in the field of manual evaluation of translated speech presentation, and hence the number of participants is relatively small given the scope of test experiments.

Related work

A broad research has been conducted during recent years. For example, Bywood et al. [2017] showed that MT is able to aid professional human translators to productivity gains, providing a promising option for partially automating the subtitling process. They discussed various aspects that MT brings, e.g low-quality

MT output or the extent of post-editing needed. Similarly, Koponen et al. [2020] studied the relation between post-editing MT subtitles and translating the subtitles from scratch. They showed that their evaluation based on spent time and the number of keystrokes indicated post-editing to be slightly better.

Even though having a whole subtitling system is beneficial in terms of time savings, individual presentation properties can be addressed separately. Matusov et al. [2019] presented an algorithm which automatically predict the end of subtitle lines given previous context. The results showed a productivity gain when using their system, compared to subtitling from scratch. Another approach was tested by Karakanta et al. [2020], when the authors investigated two methods for MT subtitling, which resulted in good performance of subtitle segmentation.

Thesis structure This thesis is organized as follows: Chapter 1 theoretically explores the presentation of subtitles, describing recommended settings for optimal usage of subtitling appearance properties, e.g. the font type and size, its colouring, and position and shape of subtitles. We continue showing the overview of used translation systems during past decades and ways how to measure their quality.

Chapter 2 is devoted to a more detailed description of our proposed implementation of the subtitling component, explaining principles of the pipeline of ASR and MT processing. The chapter also presents a design of presenting application suitable for the experiment management, maintenance and distribution among participants.

Chapter 3 examines our approach to the arrangement of experiments — the video/audio selection, requirements on participants, two main evaluation methods (questionnaires and continuous rating) and the overview of several theoretical subtitling layouts that we examine in our experiments.

Chapter 4 investigates the results of experiments. Since we use questionnaires as an additional inquiry that addresses other aspects of presentation, we compare subtitling layouts according to readability and watching comfort. We summarize the findings and recommend the usage of layouts based on given conditions.

Chapter 5 completes the thesis by analyzing collected results from the general point of view. We investigate other aspects of user satisfaction with presentation, e.g. flicker preference or measurements of latency/fluency. Finally, we relate evaluation methods that are utilized in our experiments, recommending less time-consuming evaluation given particular conditions.

We conclude the thesis by summarizing accomplished goals, leaving several remarks for a future work.

1. Simultaneous speech subtitling

Having observed the extensively spreading use of digital technologies recently, the need of acquiring as much accurate information transmission as possible while watching an audiovisual form of presentation is widely demanded. Typical situations when the complications in comprehension of the presented message occur involve an insufficient viewer's ability to perceive either the source sound or the image. The most frequent reason of this happening is a language barrier present between people with different language backgrounds.

The way how to at least partially reduce the impact of limited knowledge of the source language is to promptly deliver the piece of currently uttered speech in the form of translated text in the bottom part of a screen, known as *subtitling*. The purpose of subtitling is to serve translation for the audience that may encounter obstacles in comprehension, which may lead to spoiled experience and worse watching comfort. Moreover, subtitling is a more effortless approach to dubbing, as providing an easier access to the presented information in television programmes, shows or news for deaf or hard-to-hear viewers.

Whereas the production of subtitles by an annotator who listens to a source soundtrack and generates translated text with determined optimal time boundaries for emission is nowadays common, ways how to make the process more efficient has emerged as an attractive subject of study. Along with the improvements in the field of automatic speech recognition (ASR) and machine translation (MT), the attention of researchers inclines towards spoken language translation (SLT) and its deployment in the area of simultaneous translated speech presentation more.

In this chapter, we focus on SLT subtitling from the theoretical point of view. We subsequently describe this cross-languages presentation from the surface (the presentation of subtitles) to the deep logic (translation system). In particular, the text is composed as follows:

Section 1.1 is devoted to subtitling and serves as a guidance for the presentation of subtitles. Since recent studies have shown a number of various guidelines, concepts and conventions for a good subtitling practice, this section involves only recommendations that we follow in this thesis.

Section 1.2 provides an overview of a recent development of automatic translation systems, including several methods for its evaluation. The purpose of this section is to introduce the reader to the field of machine translation, since the resulted quality of the whole SLT presentation primarily relies on the quality of a used translation system.

1.1 Subtitling

Broad guidelines and recommendations released over the past years address various aspects of proper subtitle presentation in terms of user comfort, readability and synchronization with the given sound [Karamitroglou, 1998, Williams, 2009, Guimarães et al., 2018]. For audiovisual translation, the presentation is mostly limited to the space and time constraints. For example, Karamitroglou [1998] recommends maximum of two lines, each containing around 35 characters with the average reading speed ranging in 150–180 words per minute. Williams [2009] states that each subtitle should not exceed three lines, 34 characters per line with subtitle presentation rate no more than 140 words per minute.

However, all these recommendations are related to comfortable emission of subtitles while watching movies and cannot be directly used in simultaneous SLT because of two reasons. First, the ASR output of the processing pipeline produces unstable output which is usually further modified during subtitling (more details in Chapter 2). Second, subtitling is not restricted to the video input only, e.g. subtitling audio speech next to presentation slides may require more specific shape of the subtitling window.

Therefore, the aim of this thesis is to explore several layouts which make a trade-off between space and flicker. In other words, we examine the extent of reducing the subtitling window size at the expense of content reappearing, i.e. showing the content that have been already hidden. This includes to consider various positions and sizes of the subtitling window depending on spatial restrictions and a language source format.

In the following section, we describe general subtitle conventions which can be also used in simultaneous SLT subtitling regardless the position or size of the subtitling window. We also sketch an issue with time inconsistency which may occur during automatic subtitling.

1.1.1 Presentation

Although simultaneous speech subtitling reveals complications which are not present in offline subtitling, there are principles that hold for both approaches, namely the line control and typography. Nowadays, the majority of platforms use proportional fonts which generally differ within their character widths, thus the width of a line of subtitles cannot be defined solely by the number of characters contained. Obviously, this can be estimated experimentally, but, for now, our intention is to define it relatively given the video size.

BBC [2019], which provides detailed guidelines to subtitling, recommends to set the line width to 68% of the width of a 16:9 video and to 90% of the width of a 4:3 video. Since the 4:3 resolution is rarely used in practice, we stick to the former.

According to the typographical recommendations, we use Verdana as the font type with the size set to 8% of the active video height, which refers to the *authoring* font size. The actual, *presentation* font size depends on various ascpects, e.g. a particular device size, viewing distance, etc. It is computed as the authoring font size multiplied by a scaling factor. We set this factor to 0.53 which is picked from the recommended range 0.5–1.0, and corresponds to using a laptop or a desktop computer.

The space between lines cannot be too narrow (worse reading flow), or too wide (uselessly wasted space). Thus, the optimal line height is recommended to be 150% of the character size [Shen, 2012]. All described subtitle properties are depicted in Figure 1.1.

As previously mentioned, simultaneous speech subtitling essentially faces the issue of its intermediate instability — the output may arbitrarily change until it is finally confirmed. This behaviour can lead to continuously stacking subtitles in

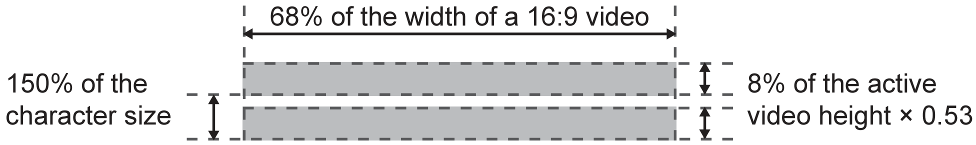

68% of the width of a 16:9 video
150% of the character size
8% of the active video height × 0.53

Figure 1.1: Two lines of subtitles which are denoted by gray rectangles, and their size defined relatively to the watched video.

the incoming buffer. It happens in situations when the invisible (already hidden) part of subtitles is updated, subtitles jump back in the history and need to recover but the scrolling speed is not set sufficiently to catch up the currently uttered word.

In addition to that, each speaker's speech differs in speed, which furthermore depends on several factors. For instance, it has been shown that stress and emotions influence the speech tempo which ranges between 3.3 and 5.9 syllables per second (syl/sec) [Arnfield et al., 1995]. The speed of speech also differs across languages, studied on eight languages with the resulting range between 5.22 syl/sec for Vietnamese and 7.8 syl/sec for Japanese [Pellegrino et al., 2011]. Similarly, there can be observed variations of the speech tempo even for different contexts and environments where a speaker is situated.

We can see that the scrolling speed depends on many factors and cannot be uniformly configured for all situations by setting one value. Therefore, it would be useful to possess a functionality that works with any source soundtrack, equally exploiting all available resources. This way, it would be ensured that subtitles are consistent with the source audio input, and are not delayed for any speaker.

We address this issue in more detail in Chapter 2 when we design our implementation of a subtitling component, which is responsible for the emission of subtitles. We show a way how to preserve a time consistent presentation by dynamically adapting reading speed.

1.2 Translation system

Although appearance characteristics of a subtitled presentation are essential for comfortable viewer's experience, the most salient part that leads to better user comprehension is the actual content of subtitles. Professional human translators are capable of making subtitles regarding the given source sound effectively, but there are situations when this approach is not applicable. For example, having an online stream the content of which has to be immediately transferred to many people who speak various languages, as many human interpreters as many target languages would be necessary to employ.

Machine translation at least partially simplifies the problem. Although the quality of the currently best translation systems is still worse than the translation of a human translator, MT may serve as a guidance for viewers who understand the source language at least partially. Thus in this section, we address MT as a way how to automatically provide translation for a source language, and show several ways how to evaluate it.

More precisely, we first present a short overview of a recent development of MT systems in Section 1.2.1. Then, we show several either automatic or manual approaches to MT output quality evaluation. Even though we do not use these methods for the presentation quality measuring in this thesis, our intention is to stress that addressing the quality of MT output is a key part of the quality of the whole SLT presentation.

1.2.1 Overview

Machine translation has gone through several stages of development in recent years. The first attempts were based on rules that were used for reordering or duplicating words, outputted according to lexical translation probabilities [Brown et al., 1993]. Additional research brought several extensions to basic word-based MT, e.g. phrase-based MT [Koehn et al., 2003], hierarchical phrase-based MT [Chiang, 2005] or syntax-based MT [Williams et al., 2016].

Increasing gain from the progression of machine learning development based on neural networks led to applying these methods in the field of MT as well, which is nowadays know as neural machine translation (NMT). Initially, recurrent networks were used as a sequence to sequence translation, afterwards encoder-decoder architectures [Sutskever et al., 2014], but the most significant improvement was achieved by Vaswani et al. [2017].

The authors of this study showed that the system of so-called *attention* or *self-attention* layers is an essential aspect how to increase NMT quality. These layers basically capture the distribution of the importance of words in the sentence, either during encoding (the traversal of source text) or decoding (the generation of translated text). Since this architecture is nowadays state-of-the-art, the translation system used in this thesis employs this strategy.

Note that the quality of SLT subtitling depends on the quality of the translation per se. Using more space for the emission of subtitles may provide a recovery from a poor MT output. Therefore, we present several frequent approaches how to evaluate MT quality, either by automatic metrics or human feedback.

1.2.2 Quality evaluation

Having a precise knowledge about the quality of MT output is fundamental in order to improve existing translation systems. There exist many methods for their rating, but we describe only the most known and widely used. The main difference between these techniques is whether they are performed automatically or a human assistance with the annotation is needed.

To address the first approach, one of the most popular automatic metrics for evaluating MT quality is *bilingual evaluation understudy (BLEU)* [Papineni et al., 2002]. BLEU is built upon the idea that the quality of MT depends on the extent of similarity between its output and the professional human translation. This algorithm always outputs a score within the range 0 (the least similar) and 1 (the most similar). The computation is performed as:

$$BLEU = BP \cdot \exp \sum_{n=1}^{4} \frac{1}{4} \log p_n,$$

where BP is a brevity penalty, the purpose of which is to penalize short translations, and p_n are the n-grams counts of the translated text found in the reference text r (or multiple texts), compared to all counts in the translated text t:

$$p_n = \frac{\sum_t \#\ \text{matching } n\text{-grams from } t \text{ in } r}{\sum_t \#\ n\text{-grams in } t}.$$

There exist multiple variations built on the principle of the n-gram or individual word comparison, such as NIST [Doddington, 2002] or METEOR [Banerjee and Lavie, 2005].

Although BLEU has been reported to be well correlated with human judgement [Coughlin, 2003], it has received several criticism [Callison-Burch et al., 2006]. Common arguments point at the fact that the increase of the BLUE score does not necessarily means the improvement of MT quality [Lin and Och, 2004]. Additionally, having the fact that BLUE is computed over a particular text, the resulted score is hardly comparable across datasets, not even languages.

Another automatic measurement of word orderliness or language fluency of a text is *perplexity*. The text is usually represented as a probability distribution over the content words and denoted as a *language model*. Perplexity expresses how large a dictionary of the language model would be if it had to select uniformly and independently for each word in the test dataset. In the information theory, having a probability distribution p of words over sentences or texts, perplexity G is defined as:

$$G(p) = 2^{H(p)} \quad \text{and} \quad H(p) = -\sum_x p(x) \log_2 p(x).$$

It is computed via *entropy* H over the probability distribution p, which is the average number of bits that we need to encode the information. Perplexity also differs across languages — richer languages in terms of the number of inflections have lower perplexity. Therefore, perplexity cannot be used as an universal evaluation method for MT quality.

The second part of the evaluation methods classification is manual assessment. There have been proposed many different manual evaluation techniques recently. They can be based on rating some quality properties of the translated text either compared to or regardless the source text. Graham et al. [2013] introduced a simple continuous scale that examines to what extent the meaning of reference text is being adequately expressed by MT output. This can be reduced to a simplified version which do not consider reference text as an input for the evaluation, and annotators are presented only a piece of source text and its translation.

The relative comparison of several candidates of different translation systems is another way how to obtain information about the translation quality. It can be performed on either whole sentences [Bojar et al., 2014] or solely highlighted segments of MT output [Graham et al., 2015]. The translation system is then ranked based on how frequently is chosen to be better than or equal to any other system.

More complex measurements are derived from the level of readers' comprehension. Callison-Burch et al. [2009] showed a method which asks annotators to edit MT output, making it as fluent as possible. Afterwards, the score is computed according to the number of edits performed. Similarly, having a context and a highlighted part of the reference text, annotators were instructed to select among several candidates, indicating which of them provides a meaning-equivalent alternative to the reference sentence [Callison-Burch et al., 2009].

Additionally, a quiz-based evaluation is an alternative to the previous two assessments [Berka et al., 2011]. Annotators are given an original source text, its translation to a target language and several yes/no questions addressing information hidden in MT output. The quality is derived from the number of correctly answered questions.

To use a manual measurement is advantageous because translation systems are made for people and only people are able to express what suits them the best. On the other hand, its disadvantage is tediousness and time-consuming process. Additionally, it is subjective and not reproducible, and may be inconsistent from different people.

In conclusion, we are aware that neither of the aforementioned evaluation techniques is used in this thesis directly, but, for the sake of completeness, we consider important to mention them. It is apparent that a MT system is a key unit of simultaneous SLT, and the methods used for its quality evaluation may be an inspiration for designing evaluation metrics of the whole translated speech presentation.

2. Implementation

The fundamental part of each simultaneous speech-translated subtitling presentation is a subtitling component which receives raw text (MT output) and emits it to users. In comparison to offline subtitling (e.g. making subtitles for movies), there are two major aspects that need to be considered when we design the presentation component for simultaneous subtitling.

First, MT is continuously fed by the last output of ASR, which makes the pipeline not stable. In subtitles, this is observed as flickering — the very last text of subtitles is repeatedly changing as machine translation alters its confidence of final hypotheses.

Second, the future speech is not predictable. In offline subtitling, time boundaries of the beginning and the end of each utterance can be easily computed to efficiently exploit space, display time and user comfort. During simultaneous subtitling, not only the speed of utterances can differ within one speech, the exact length of the utterances is not known in advance.

Having these two aspects combined together, it seems that the usage of two lines of subtitles may not be generally sufficient. Thus, it is necessary to design a subtitling component which is able to receive a user configuration in some suitable format and apply it on the subtitling window appearance and behaviour. This way, we obtain a basis for our experiments — the way how to configure various parameters to get different setups, which we compare in pairs and evaluate on user comprehension.

In Section 2.1, we describe a scalable implementation of the subtitling component which can be directly embedded to existing pipeline of simultaneous speech translation. It also supports user-friendly customizable configuration without any code exposure, which we use to create experiment layouts later.

The usage of the subtitling component is, however, insufficient to handle whole experimenting. We need another component that is responsible for the experiment management, maintenance and distribution among participants. For this purpose, Section 2.2 is devoted to the implementation of the presenting application that controls the experiment processing. Besides its practical usage, the application is intended to be easily operated by both users and us.

2.1 Subtitler: A JavaScript library

The subtitling component, denoted as a “subtitler” by Macháček and Bojar [2020], is a component which is responsible for fluent subtitle emission given an unstable MT output and limited space for the subtitling window. The *subtitling window* refers to the area where the subtitles are displayed. The authors originally implemented the subtitler in Perl, which was afterwards graphically improved and moved to web browsers as a lightweight replacement in JavaScript [Smrž, 2020]. It expects the same input format of incoming messages as in the initial design of the Perl subtitler. The same approach is used within the ELITR project[1].

In this section, we present our implementation of the subtitling component in

[1] http://elitr.eu/

JavaScript which is not only as powerful as the original subtitler in Perl, but also extended by a customizable configuration of some behavioural and appearance properties. It is designed for the web environment, which makes its usage simple and clear.

In particular, Section 2.1.1 is devoted to details of the message format that is used for communication between SLT system and the subtitler. Section 2.1.2 describes the process of data receiving, preprocessing and their textual presentation in our designed subtitling component. In Section 2.1.3, the user customization of the behavioural and appearance properties is presented in more detail. Finally, we show how much information about the presentation can be collected in the form of logs in Section 2.1.4.

Although our implementation of the subtitling component follows the same policy of the subtitled presentation as the original subtitler, we shall use the term *Subtitler* (the subtitler with the capital S) throughout the whole thesis to address specifically our implementation, preventing to accidentally interchange these two implementations.

2.1.1 Client-server communication protocol

Simultaneous SLT subtitling is consisted of a pipeline of speech recognition, segmentation, machine translation and subtitled presentation [Cho et al., 2013, 2012, Vaswani et al., 2017]. Speech utterances are automatically recognized, serving partial inputs to MT in real time. Since current MT systems internally rely on the context when a word is being translated on a particular position (Vaswani et al. [2017]), partially generated text makes the process more complicated. Subsequent addition of new words at the end of the translated text leads to frequent updates of the system confidence in final hypotheses.

The subtitling component is expected to be aware of these updates. The essential requirement on the component is its ability to tolerate that MT output is unstable. Typically, more context is provided to the system, better translation is acquired. Such a subtitling strategy that uses the notion of different states given various context lengths was introduced by Smrž [2020] and utilized by Macháček and Bojar [2020].

This strategy exploits the fact that the whole translation system is consisted of multiple components (ASR, a segmenter, MT), which are usually not deployed on the same machine. The process is decentralized and all adjacent components in the pipeline communicate through a network. The output of MT is thus sent from a server and received by a client where is processed and emitted to users.

The protocol works as following: Each sentence of MT output is classified into three distinct groups — *incoming*, *expected* and *complete*.

Incoming sentences are semantically incomplete, lacking of the final punctuation mark and determine the last sentence in the input stream. Expected sentences are created from incoming sentences by setting the final punctuation mark but the sentence content or its segmentation may be further changed by future updates. These sentences become complete when no further updates affect them. The graphical example is shown in Figure 2.1.

Since our subtitling component is supposed to be an additional possibility how to enclose the pipeline of simultaneous SLT subtitling, Subtitler uses this client-

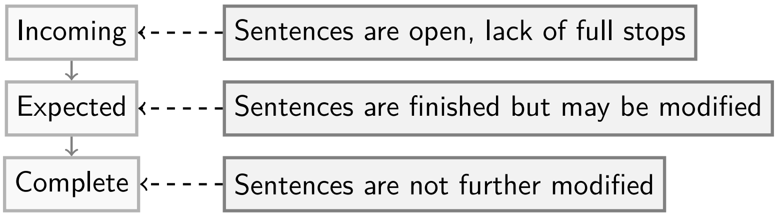

Incoming
Sentences are open, lack of full stops
Expected
Sentences are finished but may be modified
Complete
Sentences are not further modified

Figure 2.1: The states of arriving sentences. The state switch is permitted only in the direction of vertical arrows.

server communication strategy. Moreover, the component expects the exactly same format of the translation output as it was introduced by Smrž [2020], which makes Subtitler prepared to be promptly deployed as needed.

2.1.2 Data receiving and presentation

In the previous section, we described the protocol which is used for communication between the server that outputs the translation and the client that holds the subtitling component. In this section, we show how the incoming messages are received, processed and presented to users.

Each message is consisted of MT output with a unique identifier and its state. The textual content of the message is a sentence which can be characterized by one out of three descriptions from the right side of Figure 2.1. We say that every message generates at least one *Subtitler action*. Since our goal is to simplify the processing of messages as their state varies, Subtitler actions belong to one of the following elementary *types*:

- an *insertion* — translated text is appended at the end of subtitles in the subtitling window,
- a *deletion* — the part of subtitles is erased in order to be replaced by newly translated text,
- a *state change* — the switch between two states without any modification of the text, e.g. from expected to complete.

After the actions are generated, their content is applied on the subtitling window in the order of their creation. For example, replacements are formed by two subsequent actions composed of one deletion and one insertion. Even though message processing may produce several actions, they are presented as one update to users.

We can see that the working cycle of Subtitler is naturally composed of two parts — incoming data are first (1) processed and then (2) emitted to readers. Therefore, the model contains two main components: a *controller*, which handles the connection with the server and produces update information for a viewer, and the *viewer*, which takes the list of changes from the controller and manages their displaying in the subtitling window.

The window simply contains lines of words and its scrolling is controlled by an independent *refresh* which represents a fixed update, called every specified amount of time. The window also supports a system of buffering — if there is

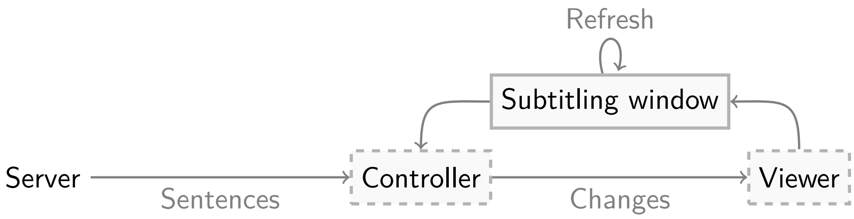


Figure 2.2: Subtitler object model. Data flow is designated in the direction of arrows.

more text received than Subtitler is able to display, the text is invisibly appended at the end of subtitles, waiting for its emergence during scrolling. The scrolling management and its configuration is described in the following section.

A desirable consequence of the Subtitler design (the split to two independent units) is that the controller cannot update the subtitling window directly, only the viewer can modify it. This approach simplifies Subtitler for both implementation and usage, which can be seen in detail in Figure 2.2.

Note that the advantage of this implementation is its flexibility. To extend the functionality it is sufficient to add a new type of action to the controller and its handler to the viewer. Or the viewer can be extended by an independent post-processing procedure, e.g. to alter visual properties of subtitles. We exactly use this option to graphically highlight the state of sentences further in this thesis (Section 3.4).

2.1.3 User configuration

Despite that some properties of a subtitled presentation can be determined for a particular group of people globally and used as a default setting (more in Chapter 5), each person is unique and prefers some details of the presentation regardless of others. Therefore, having a customizable configuration allows users to adjust the presentation properties based on their comfort.

For the sake of simplicity, we distinguish between *subtitling behaviour* and *subtitling appearance*. We start describing subtitling behaviour, which includes reading speed (fixed or adaptive), the amount of tolerated flicker and a slide up option, as can be seen in Table 2.1.

To properly select reading speed of subtitles is a widely studied area in the field of making subtitles, which we discussed at the beginning of Chapter 1. We saw that recommendations differ not only among researchers, but also between languages themselves. One way of setting reading speed is to theoretically estimate the average, resulting in 17 characters per second for Czech.[2] We refer to this speed as to *fixed* reading speed because it is independent on the actual speaker's speech speed.

Another approach is so-called *adaptive* reading speed, which we designed as a more powerful alternative to the plain fixed speed. During each refresh of Subtitler, the current value of adaptive reading speed is either (1) incremented by one if the *incoming buffer* is not empty or (2) decremented by one otherwise.

[2] The speed is derived from theoretical recommendations of words per minute and combined with the average word length in Czech. However, note that this is only a raw estimate.

Name	Description
Reading speed	Characters per second which a user reads on average. This option determines how long a certain full line has to be displayed until it slides away.
Adaptive reading speed	Minimum and maximum values for reading speed which alter the speed within the given range according to the current size of the incoming buffer.
Flicker	The user tolerance of the amount of occurred resets during subtitling, ranging between 0 (no flicker) and 1 (forced push).
Slide up	A Boolean option whether use an animation for moving subtitles up or not.

Table 2.1: User configuration properties for subtitling behaviour.

The incoming buffer includes all subtitles that are not displayed yet but are processed by the controller. The boundaries for adaptive speed are set by a user, and default on 10 and 25 characters per second for Czech.[3]

Reading speed directly influences the level of *flicker*. It is a user-specified floating point number which indicates the ratio between the number of words of stable sentences and others that are visible in the subtitling window. The flicker takes values between zero and one. Zero means no flicker, that is, all displayed sentences must be complete in order to be considered for presentation. The maximal value simply forces Subtitler to show new incoming sentences immediately (subject to queuing needed to obey reading speed), which we set as our default.

As we mentioned previously in this chapter, incoming subtitles are unstable due to updates of ASR which can eventually cause *resets* [Macháček and Bojar, 2020]. A reset shows hidden content that users already read and spoils the flow of subtitles in presentation. Thus, the flicker option is very important for the user and allows only an acceptable amount of resets.

The last parameter in the subtitling behaviour group sets whether the animation of moving subtitles up (while scrolling) is used or not. This feature is not supported in the original subtitler ([Macháček and Bojar, 2020]), which may make users harder to follow subtitles smoothly. Moreover, reset occurrences may be accidentally misinterpreted as scrolling. Therefore, we added this option to the configuration and made it default.

We continue showing subtitling appearance properties in more detail. They adjust the subtitling window width, line count, font size and word padding, and are described in Table 2.2.

We set default parameters based on proposed recommendations from Chapter 1 in millimeters.[4] Since these properties are set relatively to the video size, we avoid too complicated technicalities and do not show exact values.

[3] Boundaries are theoretically set similarly to fixed reading speed using the minimal and maximal recommended value.

[4] Setting values in millimeters is suitable for better visualization. However, note that they are later converted to pixels, which may lead to the loss of precision.

Name	Description
Width	The width of the subtitling window in millimeters.
Line count	The number of lines of the subtitling window.
Font size	The font size of the subtitles text in millimeters.
Word padding	Top/bottom, left/right padding of words in millimeters.

Table 2.2: User configuration properties for subtitling appearance.

2.1.4 Logging

Although user feedback is definitely a fundamental part of the analysis of experiments, it would be useful to have additional information about the execution of experiments. These data are typically known as *logs* and are collected as a process proceeds.

In our case, we require logs that are related to the current state of Subtitler, i.e. to its behavioral properties of the user configuration and their implications. Logs that we propose and collect during subtitling are described in Table 2.3.

As we discussed previously, the appropriately set reading speed is an important aspect of a fluent and pleasant experience of simultaneous speech subtitling. Along with the fixed reading speed we also introduced the adaptive reading speed, which determines how frequently the subtitles are moving up. Having known a current value of the speed is fundamental if we want to compare it to the fixed reading speed or simply analyze speech properties.

Another helpful information about the presentation is the amount of flicker. We defined this configurable subtitling property as a ratio between the stable and unstable part of all visible words in the subtitling window. Note that this is an approximation of the real flicker because even though some words are marked as unstable, they do not have to be necessarily replaced in future updates. Thus, when addressing flicker in terms of logging, we observe true replacements and log the count of removed words.

Similarly to flicker, we also monitor whether a reset occurs. The check is performed during each Subtitler update and suppresses updates when no resets are present. We can then investigate whether the reset occurrences are noticeable by users and whether there is any observable relation between user satisfaction and the amount of resets.

Since resets reveal a portion of subtitles that is already hidden, the side effect of this behaviour is typically a situation that the incoming subtitles are not emitted immediately. They are temporarily stored in the incoming buffer and wait until the space in the subtitling window frees by hiding older text. When this happens, having the knowledge about times when a word was received from MT and when it was displayed is beneficial. A thorough analysis of these logs may then show more precise statistics about how much the subtitles are delayed.

Lastly, we are interested in an overall visual expression of presented subtitles. Specifically, we keep track of the whole history of subtitles as they have been displayed to users. For example, we can later compute the average number of visible words at a given time, relate the value to the number of deleted words

Log type	Description
Reading Speed	Current value of the reading speed in every Subtitler update, logged as an integer in characters per seconds.
Deletions	The number of removed words during each update of Subtitler.
Resets	During each update, a short message returned if reset occurred.
Receiving time	The timestamp in milliseconds relating to every word when it was received from machine translation. Logged when the word is completed, logs of removed words are discarded.
Displaying time	The timestamp in milliseconds when a word is displayed and no further changes of future updates affect it. Logged when its status is complete.
Delay	The difference between receiving time and displaying time in milliseconds, computed for each word separately.
History	The record of history of subtitles as they were presented to the user. It captures only the final version of the text composition, intermediate steps are not included.

Table 2.3: The log types and their description. The receiving and displaying time with a delay are logged after whole line of subtitles is completed. The history is optionally downloadable after presentation ends.

and gain a true ratio to the theoretically defined flicker property. Or we can investigate a possible dependence between user comprehension and the position of words which carry the necessary piece of information.

We thoroughly examine the logs in the first part of Chapter 5. We analyze the relation between user feedback and presentation properties computed from logs, and hence obtain a better understanding of user perception of the subtitling quality.

2.2 Presenting application

In the previous section, we have introduced a powerful subtitling library, which provides a simple deployment in the environment of web browsers. The aim is to use this library not only for the subtitling itself, but also as a tool for the additional access to useful information in the form of logs. Together with the configuration of subtitling behaviour and subtitling appearance, we have obtained a core implementation for our experiments.

But Subtitler on its own is not able to manage the experiments. Thus, we need another piece of software, which is responsible for the experiment management, distribution and presentation.

The presenting application is developed in PHP upon a MySQL database.

The details of the user presentation are held by a routine in JavaScript, which internally communicates to our implemented Subtitler. Participants can log in by their unique email and a randomly generated password. The password and a short description about the application is provided via an informative email.

Since the presenting application is only a tool for the realization of experiments, we do not include all technicalities about its implementation. We only present our requirements in a list, keeping things simple and organized:

- *User feedback.* Since the main focus of the application is to monitor user satisfaction with the presentation, it has to support a functionality for collecting user feedback. We utilize two methods, namely continuous rating and questionnaires. More about them in the next chapter (Section 3.1).

- *Robustness.* We only provide the functionality which is currently allowed. For example, we disable buttons of continuous rating before the presentation starts and after it stops.

- *Platform independence.* The application allows users to watch the presentation on any device with an unrestricted operating system. Although this also includes the use of mobile phones, our requirements on subtitling layouts prohibit mobile phones because of their insufficient screen size. However, it may be added in future experiments as a possible extension.

These requirements are put on the application from the general perspective. However, designing the application requires additional aspects that have to be considered. Specifically, we have to address two essential questions.

First, how much can we ease the usage for participants? A happy user is definitely able to pay attention more than a user who is lost in the task. This is heavily influenced by the user interface which is used.

Second, how much can we ensure that the experiments are accomplished successfully? This is undoubtedly our main focus — we need to obtain reliable results which we can build upon, and hence to face issues that a simulation of an online broadcast naturally brings.

Thus, Sections 2.2.1 and 2.2.2 are devoted to answering these questions, respectively. We again present a simple list of our requirements, which we kept in mind during designing the presenting application. Lastly, in Section 2.2.3, we show a graphical example of the resulted application appearance, with a more detailed view of the experiment processing .

2.2.1 User interface

Before any experiment begins, the very first impression of each user is derived from the visual perception of the environment. It should be simple, clear and comfortable to use during annotating. The participants are supposed to get the information about their task right after they access the system. Therefore, our requirements are:

- *Simplicity.* We do not want to irritate users by overwhelmingly complicated graphical interface before they actually start to watch the videos. We also describe everything in a simple and straightforward manner.

- *Guidelines.* As long as users log in, their first glance is attracted by detailed guidance, what their task precisely is and what the interface looks like. It is also important to provide unrestricted access to the tutorial such that they can come back and read it again at any time. We include an artificial experiment to let participants try to fulfil their task.

- *Progress track.* A clear and straightforward overview of available videos with a mark, which video is already seen and which is not. This aids better orientation during the annotation as participants always know what portion of the videos they already watched.

- *Clarity.* All components needed by either the experiment management or the presentation per se are visible at once. Therefore, the users have an opportunity to adjust the view scale according to their comfort, to which they were explicitly encouraged.

User satisfaction with the usage of our presenting application is a key aspect to obtain precise results of the experiments.

2.2.2 Stream stability

The main goal of our experiments is to investigate various subtitling layouts of an online broadcast. Our experiments relies on stream stability either when the source soundtrack is being loaded or when the user rating is being collected. An additional aspect of a successful online simulation is a prevention of accidentally interrupting the data transmission. More specifically, we require:

- *Connection independence.* Since a real presentation can crash on a bad Internet connection, considering all possible technical issues is beyond the scope of our experiments. Thus, we ensure that the simulated stream is stable by preloading source videos to the user's browser.

- *Local backups.* Since the Internet connection is usually unreliable, we keep a copy of user feedback as a local backup on the user's browser to avoid data lost. When a connection corruption occurs, users are gently warned after watching that they are obliged to download the feedback data and send it manually to us.

- *Avoiding interruptions.* Simulating online speech reliably, that is, prevent users to pause the presentation by any way. We disable media controls while the presentation is running. Closing the browser or refreshing the page is possible but unfortunately unavoidable.

We believe that all these requirements lead to a fluent and reliable simulation of an online broadcast, which do not spoil results of our experiments.

2.2.3 Usage

A simple diagram that summarizes the usage of the application and captures some of its design properties is shown in Figure 2.3. It describes steps that participants have to follow to successfully complete the task. After logging, they are

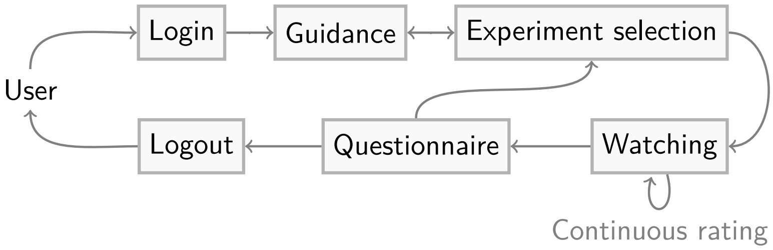


Figure 2.3: The usage of the presenting application in a diagram.

presented by a guidance which shows the usage and explains their task. They can move to the experiment selection, which redirects them to the chosen experiment. During watching, their continuous feedback is not only sent to the server, but also collected on their machine locally as a backup. Participants finish the experiment by filling in questionnaires. The process is completed by selecting another experiment or returning back to the guidance.

We can see that the process is simple and straightforward. In the following chapter, we examine the experimenting in more detail. We explain our approach to the creation of experiments, describe evaluation methods thoroughly, and show our requirements on the selection of participants and documents.

3. Experiments

In Chapter 1, we discussed subtitling from the general point of view and presented recommendations for displaying subtitles. We additionally mentioned that having increased popularity of MT, a possible extension of offline subtitling is to simultaneously translate the speech and make subtitles instantly. We showed a brief overview of developing translation systems during last decades, describing one which is nowadays state-of-the-art and which is used for experiments in this thesis. We also presented few automatic and manual evaluation techniques for rating these systems.

Afterwards in Chapter 2, we described an implementation of a JavaScript subtitling library called Subtitler, which was developed upon an already existing pipeline of automatic speech recognition, segmentation and machine translation. The subtitling component utilizes all advantages of its two predecessors and is additionally extended by new features, e.g. adaptive reading speed or a slide up animation. We also introduced a presenting application which was created for the simulation of online streaming locally on the user's machine, aiming at convenient management, maintenance and distribution of experiments.

The recent text was a preparation to the main goal of this thesis — to experimentally investigate the quality of subtitled presentation which is evaluated by humans given different constraints. It is a huge task where numerous aspects have to be taken in account — which subtitling layouts are used, which properties of the presentation are investigated, what are our assumptions about the user preference, how the experiments are built and, most importantly, which measurements are utilized for the evaluation of user comprehension.

In this chapter, we show our approach how to build and evaluate experiments. Specifically, in Section 3.1, we present two evaluation techniques that we use in our experiments. In Section 3.2, we describe the process of a document selection and formulate our requirements. Section 3.3 is devoted to participants who rate the quality of presentation, and we sketch our assumptions about their comprehension based on different source language proficiency.

In the last section of this chapter, in Section 3.4, we present several theoretical subtitling layouts which we examine in our experiments. We also formalize our assumptions about their advantages and disadvantages, which are later used as a basis for analyses.

It is important to mention that participants are prohibited to modify the parameters of Subtitler during experimenting, the user configuration serves as a tool of creating the experiments. The appearance and behaviour properties are used in their default form as we described in Section 2.1.3.

3.1 Evaluation methodology

There exist several approaches how to automatically or manually evaluate MT output, some of them were presented in Chapter 1. However, our experiments require a more sophisticated evaluation because the overall quality of the MT output is additionally influenced by online streaming and by the selection of a subtitling layout.

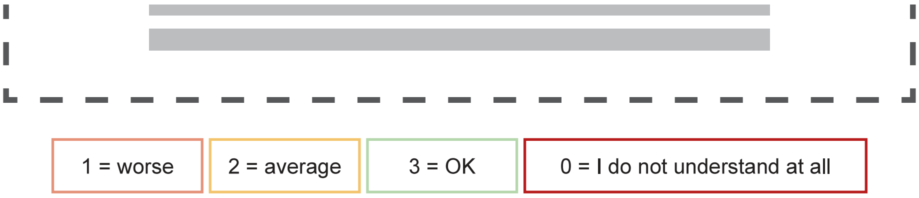


Figure 3.1: A graphical example of continuous rating buttons below a video which is overlaid by two lines of subtitles. The labels are originally in Czech, i.e. in the native language of participants. The buttons also react on Czech keyboard when pressing special characters of its top row.

Thus, to evaluate simultaneous SLT subtitling given various conditions, we present two evaluation methods on user comprehension — continuous rating and questionnaires. The former uses subjective user assessment and was originally introduced by Macháček and Bojar [2020]. The latter addresses subtitling quality from the objective perspective, uses factual questions about the presented content and is inspired by Berka et al. [2011].

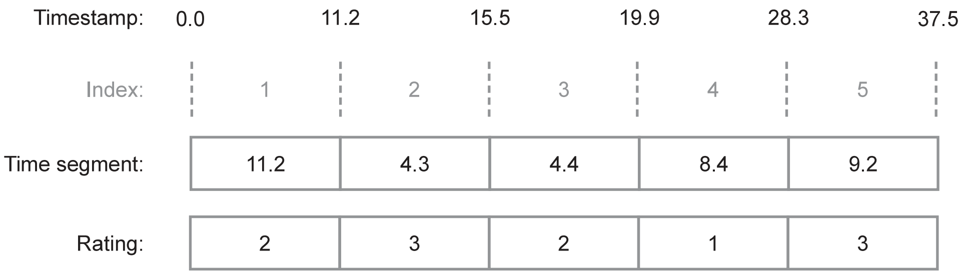
In the following section, we demonstrate the process of participant voting in more detail, employing the continuous rating evaluation with a description of its inner implementation and graphical view. We formulate benefits of this evaluation, and mention situations where the continuous rating may be inadequate and may skew results.

3.1.1 Continuous rating

The first evaluation that we employ is inspired by Macháček and Bojar [2020]. During the presentation, participants are asked to continuously press one of four buttons, depending on their current comprehension level. The buttons are labeled by a number within the range 0–3 with an additional short textual description. Zero means that subtitles are not understandable at all and three means the opposite, that is, the content of subtitles is conveyed without any complications.

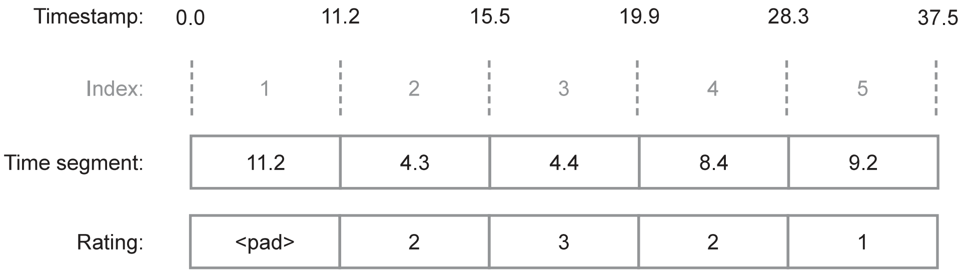
The buttons are positioned either below the subtitling window or below a video image, depending on which layout is used (more about layouts in Section 3.4). Participants have two possible ways how to express their comprehension — either with a click of the mouse or by pressing one of four keyboard keys. Since we assume that the use of keyboard is more convenient, we adjusted the order of buttons on a screen such that it matches the order of keys on the keyboard, and encouraged participants to use the keyboard. A graphical illustration of the buttons of continuous rating is shown in Figure 3.1.

When pressing a button, a pair of two values is recorded. Every pair consists of one timestamp when the action was performed, and one rating value that corresponds to the numeric label of the pressed button. The timestamp is further processed and only the difference between two adjacent timestamps is stored — let us denote these time differences as *time segments*.



Timestamp:	0.0	11.2	15.5	19.9	28.3	37.5
Index:	1	2	3	4	5	
Time segment:	11.2	4.3	4.4	8.4	9.2	
Rating:	2	3	2	1	3	

Figure 3.2: The illustration of the first few records when the buttons of continuous rating were pressed. The numerical values of timestamps and time segments are stated in seconds, but they are originally measured in milliseconds for better precision.



Timestamp:	0.0	11.2	15.5	19.9	28.3	37.5
Index:	1	2	3	4	5	
Time segment:	11.2	4.3	4.4	8.4	9.2	
Rating:	<pad>	2	3	2	1	

Figure 3.3: The correction of an inaccurate pairing of time segments and rating values. The description of numbers is the same as in Figure 3.2.

We thus collect two sequences, one for time segments and one for rating values, which can be seen in Figure 3.2. However, the elements of the sequences are not properly aligned, i.e. it does not hold that during the i-th time segment user comprehension corresponds to the i-th rating value. The reason is that participants click on the button when the assessment changes, so the currently ended time segment is related to the previous rating value. In order to work with correctly element-wise paired sequences, we simply use a special value as a padding in the beginning of the rating sequence. This minor modification is captured in Figure 3.3.

The salient advantage of the continuous rating is the simplicity of preparation. Once the user interface for the rating is implemented, it can be used for any video or audio independently. Additionally, we obtain a partial evaluation for each time segment, which can be later analyzed individually. The boundaries of the segments are determined by participants themselves, so the choice of the pressing frequency is based on their comfort.

On the other hand, participants may so deeply concentrate on reading subtitles or following the sound or a video image that they forget to press the buttons. Another disadvantage is the unreliability of this evaluation — subtitles may be perfectly grammatically correct but the translation would be wrong. In other words, the user understands but it is not what has been really said.

3.1.2 Questionnaires

To address the issue with possible inadequacy of continuous rating, we propose an evaluation which is based on prepared factual questions about the spoken content. Particularly, after watching a video or listening to an audio (let us shortly refer to a *document*), each participant is given a questionnaire which addresses two main regions:

1. a *comprehension level* — an ordered list of factual questions that are constructed within around every 30 seconds of the input sound and are unique for each document, and

2. *general feedback* — a group of general questions that measure the watching experience from different aspects and are common for each document.

The former is primarily focused on an unbiased evaluation of user comprehension. The majority of questions were created in open-ended manner, that is, the questions required participants to write a more complex and descriptive answer instead of providing a simple yes/no response.

In particular, the questions were prepared by a Czech teacher of German who also provided an answer key regardless of the MT output. The questions were selected in such a way that the answers were known only after watching a document, and not from the previous knowledge. The questions were expected to be easily answerable for a native speaker who had graduated from a high school. We thus avoided technicalities or special terms, which would have required some additional education.

In addition to that, each factual question was followed by a complementary inquiry whether the knowledge came from subtitles, a video image, sound or whether the answer had been known before the experiment began. We also gave a participant an opportunity to select an option saying that he or she forgot the answer.

The latter part of questionnaires investigates the presentation of subtitles from the general perspective. We asked participants about their satisfaction with fluency, latency, readability, ability to follow a video image if included, adequacy and overall impression of the subtitles. The exact wording of all general questions is captured in Table 3.1.

To evaluate the correctness of factual questions and hence get the overall score, we divided answers into five categories according to the answer key — correct (OK), partially correct (OK-), wrong (WR), unknown (UNK) and forgotten (FG). An intuitive way how to compute the overall score of the experiment would be to count all correct and partially correct answers and express this number as a fraction regarding to all questions assigned to the used document. This way, we would obtain a measure which is comparable among all tested experiments.

However, it is straightforward that this method not only does not consider all categories, but is also not extendable. It is probably more appropriate to penalize partially correct answers, so they lower the overall rating, unlike to keep them at the same level with the correct ones. Or wrong answers can be considered as harmful. Thus, we propose a method which assigns some decimal *weight* between zero and one to each category. The overall score is therefore computed as a weighted average of all categories.

Area	Question
Fluency	How do you rate the subtitles in terms of fluency or language accuracy?
Latency	How do you rate the subtitles in terms of synchronization with the source?
Readability	How much did you manage to read all the subtitles?
Watching	If the document contained an image, how much did you manage to watch it?
Adequacy	How much do you think you were able to know when the translation was inadequate?
Total	What was your overall impression of the subtitles?

Table 3.1: The overview of general questions which were added and the end of each questionnaire.

Since the weights can be assigned to the categories in many different ways, for the sake of simplicity, we move the discussion about the selection of weights to Chapter 5, where we also analyze the evaluation empirically.

We can see that weighting makes the evaluation using questionnaires scalable and more precise (in terms of true correctness) than continuous rating. On the other hand, it is very time-consuming — not only preparation of all questions, but also its labeling into categories costs time which increases in the size of a document collection.

3.2 Documents

Overall comprehension of simultaneous SLT subtitling depends on many distinct factors, but mostly on an input sound quality, a video image and a presented content (a topic, informativeness, interest). Since the source of a speech may be either a video or an audio, we generally denote each speech source as one *document*. The documents have properties, which we use for dividing the documents to groups according to some criteria.

Since our goal is to use a human evaluation system, we need to ensure that we select such documents that lead participants to watch the presentation fully focused with as little distraction as possible. In order to achieve this, we choose documents that satisfy the following conditions:

- Documents are of a *feasible length*. Having too long documents may not only bore participants, but also make participants forget salient information from the beginning of documents and spoil the results of questionnaires. Therefore, we aimed at the target document duration ranging between 5 and 10 minutes (except few cases).

- To prevent participants' fatigue even more, we choose documents which are *interesting* for their content, e.g. current news (an illustrative video about

Source	Type	Count	Length	avg±std
Maus: For children’ education	V	2	14:43	7:22±221
Dinge: For teenagers’ education	V	2	16:09	8:05±226
DW: For inter. learners of German	A	2	18:48	8:34±150
Mock interpreted conferences	A	3	27:52	9:17±058
DG SCIC: For interpretation training	TP	3	17:34	6:24±141
European Parliament	TP	3	18:08	6:03±055
All		15	114:52	7:33±135

Table 3.2: The Source of selected documents. *Type* denotes audio only (A), talking person only (TP) and video (V) with illustrative or informative content. The length and the averaged length is reported in minutes and seconds, the standard deviation only in seconds.

vaccines or about the quality of different respirators), politics (European Union and its benefits) or the history (a dark past of a Berlin charity).

- To avoid information forgetting, we focused on documents that are *easily understandable*, do not require any previous special knowledge and do not contain the usage of technical terms.
- We additionally exclude documents whose translation is not of *representative quality* because of unusual anomalies such as isolated utterances with long pauses or many named entities.

For the sake of experiment purposes, we also consider limitations which are more related to experiment technicalities and goals, rather than participants' focus:

- It is important to choose a proper source language. We choose German because it is not so popular and hence we can find more easily participants who do not speak German at all. This is not the case for English, which is nowadays widely spread.
- We select documents of different visual *type*, that is, audios (A), videos with an graphically illustrative content (V) and video of a talking person only (TP). We may thus test different subtitling layouts based on a various source of information.

A brief summary of selected documents is described in Table 3.2. We used 15 documents: 6 videos without any illustrative content (talking person only), 5 audios and 4 illustrative videos. The duration of all documents is almost 115 minutes and the average document length is 7 minutes and 33 seconds.

Level	Count	Group	Total
None	5	non-German speaking	10
A1	5		
A2	1	German speaking	4
B1	2		
B2	1		

Table 3.3: The assignment of participants to groups based on their German knowledge level. The proficiency is reported by CEFR scale.

3.3 Participants

An intuitive requirement on our experiments is that participants should not understand the sound since our goal is to evaluate simultaneous SLT in different presentation settings. However, the level of language proficiency may differ and, in fact, this is a very interesting question in simultaneous SLT presentation: Is the preference of the subtitled presentation similar or distinct among users with a different source language understanding? Thus, we selected not only subjects who do not understand German at all, but also subjects who have some experience with this language.

Specifically, the sample of participants was consisted of 14 native Czech speakers who were asked to fill a detailed form about their knowledge of German. The questions were related to their experience with German (if they had some), for instance when they began to learn, how long they learned or how long they actively use German. The level of proficiency was reported in CEFR[1] scale, which divided participants into groups of the level ranging between zero and B2. We did not allow the level to exceed B2 because we needed participants to be at least partially dependent on subtitles and the translation. We also ensured that they do not have any knowledge of other languages which might help with their comprehension.

3.3.1 Source language knowledge

Taking into account that the size of our sample is only 14 and and the number of groups of different proficiency levels is relatively large, the distribution of participants is very sparse, as we can see in Table 3.3. Therefore, for further analyses, we decided to merge some groups and work with only two of them, which we denote as *non-German speaking* and *German speaking*. The pairing of the proficiency level and the group name is captured in Table 3.3 as well.

By making this distinction we suppose that these groups have different preferences on subtitle presentation, namely for the flicker option, e.i. how much the subtitles are delayed during the presentation. We hypothesize that participants who understand the source language at least partially prefer subtitles to be as up-to-date as possible because they follow the sound and subtitles concurrently.

[1] Common European Framework of Reference for Languages

On the other hand, participants who do not understand the source language at all or very minimally prefer stability before smaller delay.

3.4 Subtitling layouts

To find a proper balance between space and flicker, we investigate several possible *layouts*. A layout is a combination of spatial, positional, typographical and visual characteristics of presented subtitles. Each layout is supposed to be beneficial regarding to a specific situation. For instance, if the subtitling window is short and narrow, all its content is always up-to-date, which aids to concurrently follow a video image and subtitles. On the other hand, a small window contains short history, so the user can miss translated content if it disappears while paying attention to the video.

Similarly, the user preference may depend on the source of speech. When the sound is not presented with a video image, i.e. the user listens to a radio broadcast or any other type of the audio source only, spatial constraints of the subtitling window size are relaxed and the number of subtitle lines may be increased. This way, a user is able to quickly seek the history if he or she miss the translated content.

For the sake of clarity, we distinguish between three basic layouts based on the position of the subtitling window — *Overlay*, *Below* and *Left*. We thoroughly describe their properties in the following sections. We also present our observations about their possible advantages and disadvantages, according to which we formulate formal hypotheses that interleave the text. Then, we analyze the experiments that employ different layouts in Chapter 4, where we also show whether the hypotheses are empirically proven or denied.

By the end of this chapter, we describe an additional layout which addresses visual representation of a flicker state. Its purpose is to guide users to make a decision on which parts of subtitles they focus on according to their tolerance of the output instability. We denote this layout as *Flicker Highlighting*.

The majority of videos are in the 16:9 format, so we display all videos in the 16:9 frame. Audios are presented as wide narrow rectangles which contain media controls and the current position in the played sound. This composition is used in all layouts that are described in the following.

3.4.1 Overlay

Emitting subtitles over a video image is probably the most traditional way how to balance the available space and user comfort while reading the subtitles. Users do not need to make a great effort to switch the focus from a video image to subtitles because their eyes tend to overcome a very small distance. BBC [2019] recommends to type each word in white text on a rectangle fully filled with black, which helps to visually blend subtitles into a video, especially for darker video backgrounds. Watching the image thus become more comfortable in terms of less distraction caused by subtitle changes.

On the other hand, designers advise to avoid high contrasts between white text and its dark background [Anthony, 2020, Optical, 2020]. It causes that the eye works harder since it opens wider because of absorbing more light. Then, the

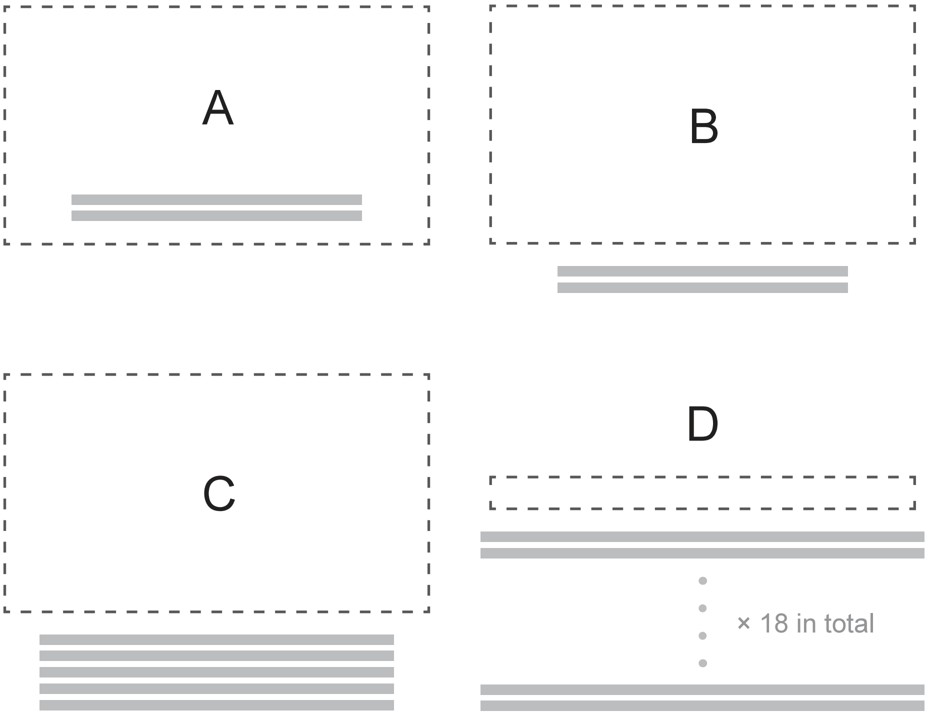


Figure 3.4: The subtitling window displayed over a video image in Layout A (upper left), and below a document in Layout B = Small (upper right), Layout C = Medium (lower left) and Layout D = Large (lower right). The audio is represented by a narrow dashed rectangle in Layout D. To show an overall visual impression of all layouts, the ratio between the subtitling window width and the video image size is preserved.

white letters can bleed into the black background, which makes the text blurry. This is also known as the halation effect [Anthony, 2020].

Since the use of white text on a black background is nowadays common in video subtitling, we follow the manual but alter the contrast to be more acceptable for the readers' eyes. We set the opacity of the background rectangle to 0.7 for better subtitle perception and readability. We also use two lines of subtitles, each of the length of 68% of the video width. Together with all font properties that we described in Section 1.1.1 this layout refers to defaults from BBC [2019]. It is illustrated in Figure 3.4 as Layout A and denoted as *Overlay.*

We suppose that having only a traditionally provided space for the subtitle emission, namely two lines of the width less than the video size, the most preferred way is to position the subtitling window over the video image.

Hypothesis 3.4.1 (Overlay & Comprehension)**.** *When restricted to two lines of subtitles, the best comprehension is acquired by using the Overlay layout.*

3.4.2 Below

The purpose of the Overlay layout is to equally exploit space, subtitle readability and image watching. However, automatic simultaneous speech subtitling may require further space for the subtitling window in order to keep the stable content displayed for enough time such that users can comfortably read it. Extending the size of the subtitling window that overlaps the video image more may spoil the overall impression of the video even though the space for subtitles is increased. Thus, we propose to move the subtitling window below the video where more lines of text can be reserved. We denote this layout as *Below*.

Since subtitles are no longer emitted over a video image, we use black text on a white background instead of the opposite. This is also beneficial from another point of view — studies have shown that subjects read with 26% greater accuracy when they read with dark characters on a white background [Bauer and Cavonius, 1980].

Hypothesis 3.4.2 (Below & Readability)**.** *Subtitles emitted below a video image are more readable than emitted over the image in the Overlay layout.*

As mentioned above, moving subtitles below a video allows the subtitling window to be of the size such that does not interfere with the visual content of the video. We suggest three variants of this layout, which differ in the subtitling window width and in the number of subtitle lines. In particular, we consider the Small, Medium and Large type, each providing unique characteristics for usage.

Small The most basic type. In comparison to the Overlay layout, this layout differs only in the position of the subtitling window, which is situated below the video. It contains two lines with 68% of the video width (Figure 3.4, Layout B). When comparing to larger subtitling windows, this type may benefit from its compact size and fast user focus switching, reflecting on the user watching comfort and readability. The potential disadvantage is short history, which may reflect on user comprehension.

> **Hypothesis 3.4.3** (Below Small & Watching comfort)**.** *The watching comfort of the Below Small layout is higher compared to layouts that employ larger subtitling windows, especially for illustrative videos.*

> **Hypothesis 3.4.4** (Below Small & Readability)**.** *Readability of subtitles in the Below Small layout is better compared to subtitling windows that contain more lines, especially for illustrative videos.*

Medium This layout, referring to Layout C in Figure 3.4, is an extension to the Small layout in both the window width and the number of subtitle lines. Particularly, we add three lines and we decrease the *margin* of the subtitling window to its half, where the margin refers to a free space of both sides of the subtitling window (until the video size is reached). The new subtitling width then corresponds to 83% of the video width. We suppose that a larger subtitling window helps to catch more information from the presentation as a result of less resets. On the other hand, it may harm the watching comfort.

> **Hypothesis 3.4.5** (Below Medium & Comprehension)**.** *The user comprehension while using Below Medium is better than while using smaller subtitling windows.*

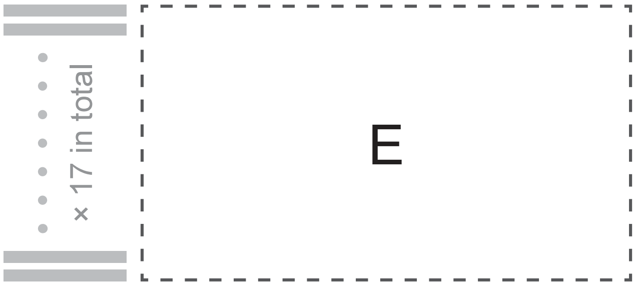


Figure 3.5: Subtitles emitted on the left side of a video. The example graphically preserves the ratio between the video and the subtitling window size.

Large In order to transfer as much information stably as possible, we create a subtitling window whose size approximately fits the size of the video. Specifically, this layout is consisted of eighteen lines, the width of each is equal to 105% of the video width. A graphical example can be seen in Figure 3.4, denoted as Layout D. Since the subtitling window is relatively large in comparison to the video, we restrict its usage to an audio source only. We assume that using this layout is suitable especially for users who prefer stability, which increases their comprehension and improves the readability of subtitles.

Hypothesis 3.4.6 (Below Large & Comprehension)**.** *The Below Large layout achieves higher user comprehension compared to layouts that employ smaller subtitling windows.*

Hypothesis 3.4.7 (Below Large & Readability)**.** *Below Large benefits from better readability compared to layouts of smaller subtitling windows.*

We hypothesize that emission of subtitles below the video helps users to conveniently read subtitles while focusing on the translation more than on the video image. Especially, this is the case when the video image do not carry any visual information or is highly limited.

3.4.3 Left

Despite that the Medium Below layout is devoted to better readability of the stable part of subtitles, users who prefer lower latency have to focus on bottom parts of subtitles. Then, there is a gap between the video image and the text that is being read, which may lead to a decrease of the users' happiness.

Therefore, we propose a layout where the subtitling window is moved to the left side of the video. The width of the subtitling window is set to 25% of the video width and its height fits the video, which results in seventeen lines. A graphical example is in Figure 3.5.

This layout is advantageous in two aspects. First, focus switching between subtitles and the video image is faster. Second, longer history is exposed and hence this layout satisfies needs of users who prefer stability.

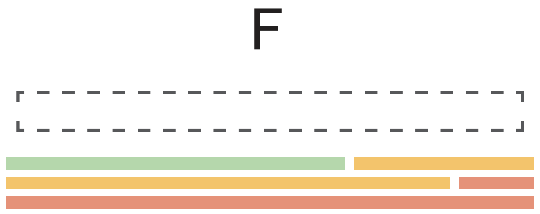


Figure 3.6: Three lines of subtitles below an audio source. The first visible part of the text background is green (complete sentences), which is followed by orange (expected sentences) and ended by red (incoming sentences). This order is always preserved, although some parts may be missing, e.g. complete sentences are directly followed by incoming and so on.

Hypothesis 3.4.8 (Left & Comprehension)**.** *Thanks to its longer history, the Left layout ensures increased comprehension if we contrast it to the Below Small layout.*

Another advantage is that while reading one line of subtitles, users can naturally continue with their look in the direction of reading[2] and see the current video image. On the other hand, short lines contain much less words and may spoil the flow of smooth reading.

Hypothesis 3.4.9 (Left & Watching comfort)**.** *Watching a video image is more comfortable while using the Left layout rather than using the Below layout.*

Hypothesis 3.4.10 (Left & Readability)**.** *The Left layout harms the quality of subtitle readability.*

3.4.4 Flicker Highlighting

In Section 2.1.1, we described a communication protocol where each received message is classified to one out of three labels — incoming, expected and complete. Whereas complete sentences are fully finished, the segmentation of expected sentences may change, and incoming sentences may change even more because they represent the bottom of the incoming buffer. The idea is to exploit this information and represent each state graphically.

The layout that we introduce is not restricted to any specific size or source type, it only prescribes that messages are highlighted regarding the state they belong to. In particular, each word of a message is emitted with a coloured background — a green colour determines the complete state, an orange colour stands for the expected state and a red colour is assigned to the incomplete state. A graphical illustration of the layout can be seen in Figure 3.6, and we denote it as *Flicker Highlighting.*

We suppose that users intuitively notice the relation between colours and states, and focus on parts of subtitles that satisfy their needs and hence understand the presented content better. Note that this works even without exact

[2] Since Czech is our target language which follows left-to-right writing, we do not consider other languages with different directions of writing.

knowledge of the inner implementation because the colour–state pairings are chosen similarly to how humans are used to associate colours with objects in everyday life, e.g. traffic lights. Thus, this layout is assumed to be beneficial for any user preference.

Hypothesis 3.4.11 (Flicker Highlighting & Comprehension)**.** *Flicker Highlighting aids in better understanding subtitles.*

We also hypothesize that the visual representation of a flicker state aids in readability because users know what parts of subtitles are stable.

Hypothesis 3.4.12 (Flicker Highlighting & Readability)**.** *Subtitles emitted with highlighted words regarding their flicker state are more readable.*

4. Subtitling layout analysis

The quality of the subtitled presentation is clearly linked to given spatial restrictions. In Section 3.4, we have introduced several possible theoretical subtitling layouts whose advantages and disadvantages depend primarily on the provided space, the type of the source sound (video or audio) and its visual informativeness (whether a video is illustrative or not).

In this chapter, we examine the preference of users regarding some setting in which the subtitles are presented. By the term *setting* we understand a unique combination of parameters which is distinguished from other settings at least in one property. For instance, one setting would be the Below layout with the subtitling window size of 2 lines with 167 millimeters, emitted on white background. Another setting would have the same properties except its subtitling window size, which would change to 5 lines with the width of 200 millimeters. We also define *an instance* which is a setting extended by a document and participant. In other words, it is the realization of a setting for a single participant.

In Table 4.1, we present an overview of settings which were created as a basis for our experiments. *Layout* refers to a theoretical layout that we described in Section 3.4. *Type* denotes the target type of a document for which the setting is made. This restriction stems from the fact that some properties of subtitling layouts, e.g. the shape or size of the subtitling window, are not suitable for a video because they simply do not fit the user's screen. *Colour* describes the usage of word state highlighting by using three background colours, which represent three basic states of MT output — complete, expected and incoming. Despite that the size of the subtitling window and the document type is not restricted in the theory, for the sake of simplicity, we utilize only settings with the height of 18 lines and the width of 250 millimeters for the audio input only.

We also added concrete sizes of the subtitling window, which are denoted in the format (number of lines)×(width in millimeters), and are depicted in the *Size* column. We report width in millimeters for a better visual imagination, but recall that the subtitles are presented in a web application where the participants are encouraged to customize the scale (as we discussed in Section 2.2). Thus, the size is only a reference value which may further change, but it is important to point out that ratios between any two components of the presentation are always preserved.

Finally, *Experiment* captures names of investigated experiments, each dealing with two contrasting settings that were created to address some of our hypotheses about the users' preference (the hypotheses were discussed in Section 3.4). It is also important to mention that during the analysis of every experiment we selected instances of settings which were not underrepresented by documents, i.e. for each experiment we chose only these instances which have at least one common document in both compared settings.

Generally, the last three columns represents essential parameters that were combined while creating settings. Note that not all parameters can be combined arbitrarily — for example, using the subtitling window on the left side of the video, we have to use the subtitling window size of 17x60 as well.

The preference of subtitling layouts can be measured by multiple ways. Be-

Experiment	Layout	Type	Colour	Position	Size
Overlay vs Below	**A**	**V/TP**	**No**	**Overlay**	**2×163**
	B	**V/TP**	**No**	**Below**	**2×163**
Below vs Left	B	V/TP	No	Below	2×163
	E	**V/TP**	**No**	**Left**	**17×60**
Small vs Medium	B	V/TP	No	Below	2×163
	C	**All**	**No**	**Below**	**5×200**
Medium vs Large	C	All	No	Below	5×200
	D	**A**	**No**	**Below**	**18×250**
Flicker highlighting	D	A	No	Below	18×250
	F	**A**	**Yes**	**Below**	**18×250**

Table 4.1: The overview of settings that we used in experiments. *Layout* refers to theoretical subtitling layouts from Section 3.4. *Type* representation is the same as in Table 3.2. The subtitling window size is in the format (lines × millimeters). Unique settings are bolded.

cause the nature of any presentation which includes a natural language is to deliver some information to users, our main evaluation metric is the level of user comprehension. We have presented two approaches in Section 3.1, namely questionnaires and the continuous rating. We focus on this analysis in the Section 4.1.

In addition to comprehension, we also evaluate the presentation of subtitles by the level of users' satisfaction with the readability and watching comfort. We obtained the scores from the general part of questionnaires and we describe the results in Section 4.2 and Section 4.3, respectively.

Since the sample of non-German speaking participants is too small, we do not split the participants to two groups according to their language level proficiency as we discussed in Section 3.3. Additionally, none of the results is statistically significant (according to two-sided t-test), which we explain by insufficient size of testing sample.

4.1 Comprehension

The level of user comprehension is essential when deciding which subtitling layout is suitable for the presentation. Discussions about which layout is beneficial in which situation were held in Section 3.4. Basically, our assumptions mainly depend on the length of the subtitle history because earlier emitted, more stable output of MT is displayed longer.

In this section, we thoroughly analyze experiments from Table 4.1. During each analysis, we state our assumptions, describe the results and formulate conclusions whether the experiments empirically confirm or deny the hypotheses.

Since manual evaluation of factual questions is more reliable than self-judged

Evaluation	Type	Below		Overlay	
Questionnaires	talking	9	0.31 ±0.27	9	**0.38** ±**0.22**
	video	5	0.22 ±0.12	8	**0.34** ±**0.10**
	sum, avg	14	0.28 ±0.24	17	**0.36** ±**0.18**
Continuous rating	talking	9	1.65 ±0.52	9	1.65 ±0.99
	video	5	1.11 ±0.50	8	**1.15** ±**0.77**
	sum, avg	14	**1.47** ±**0.57**	17	1.42 ±0.93

Table 4.2: User comprehension of Below vs Overlay experiment. The three numbers in each row and cell are the number of experiments, average and standard deviation. The higher score, the better. The questionnaire evaluation is between 0 and 1 and continuous rating is between 0 and 3. Higher score in each row is bolded.

continuous rating, we will focus on results of the questionnaire evaluation more. Nevertheless, reporting both evaluations is suitable for their comparison and we investigate this relation in Section 5.2.2 in more detail.

4.1.1 Overlay vs Below

We begin with the basest experiment which compares the Overlay layout with the Below layout. In Table 4.2, we can see the results of continuous rating and questionnaires. We divided the results according to the type of a document to videos of a talking person and videos with an illustrative content. The third row in every group captures a situation when the observations are not split.

The questionnaire evaluation shows a clear evidence of better comprehension when using Overlay than Below — it is easier to follow image and text simultaneously if there are obstacles with understanding the speech. On the other hand, continuous rating reports a minor benefit of the Below layout, but considering only illustrative videos the inclination is towards the Overlay layout. Let us recall our assumption:

Hypothesis 3.4.1 (Overlay & Comprehension)**.** *When restricted to two lines of subtitles, the best comprehension is acquired by using the Overlay layout.*

Overall, the results confirm our hypothesis that having only traditionally allowed space for the subtitling window its preferred position is over the video image. This way, the watching and reading aid to spot more critical parts of the subtitles, which reflects on the amount of understood content.

4.1.2 Below vs Left

Similarly, the next experiment is devoted to the comparison based on a different subtitling window position, but now we contrast the Left layout and the Below layout. The results are again split into two groups based on the nature of a document, and can be seen in Table 4.3.

		Subtitling window position	
Evaluation	Type	Left	Below
Questionnaires	talking	5 **0.36 ±0.28**	7 0.31 ±0.29
	video	1 0.20 ±0.00	3 **0.30 ±0.07**
	sum, avg	6 **0.33 ±0.26**	10 0.30 ±0.25
Continuous rating	talking	5 1.56 ±1.00	7 **1.78 ±0.35**
	video	1 0.23 ±0.00	3 **1.21 ±0.45**
	sum, avg	6 1.33 ±1.04	10 **1.64 ±0.45**

Table 4.3: The comparison of the Left and Below layout from the perspective of user comprehension. The number representations are the same as in Table 4.2.

In this case, while evaluating questionnaires, comprehension is better for videos of a talking person in the Left layout and for videos with an illustrative content in the Below layout. The results show that we were partially correct — we supposed that longer history in the Left layout helps with comprehension due to reduced flicker, which we formalized in the following hypothesis:

Hypothesis 3.4.8 (Left & Comprehension)**.** *Thanks to its longer history, the Left layout ensures increased comprehension if we contrast it to the Below Small layout.*

This holds for videos of a talking person where the focus switching on a video image is not present. On the other hand, when a video provides additional information in the image, the participants understood more when the Below layout was used.

Continuous rating reports better scores for all groups in the favour of the Below layout, even for videos of a talking person. Altogether with the results from the questionnaire evaluation of videos, we conclude that the unusual narrow shape of the subtitling window in the Left layout generally harms user comprehension. However, its usage may be beneficial for non-illustrative videos.

4.1.3 Video: Small vs Medium

Table 4.4 (upper) summarizes the level of comprehension while altering the subtitling window height. More precisely, in this experiment we investigated theoretical layouts Below Small and Below Medium. We supposed that longer history helps better information acquiring because there are less reset occurrences and users can easier follow the stable part of MT output:

Hypothesis 3.4.5 (Below Medium & Comprehension)**.** *The user comprehension while using Below Medium is better than while using smaller subtitling windows.*

We can see that we obtained better comprehension in the case of the longer subtitling window when evaluating factual questions. Specifically, the highest improvement is reported for videos of a talking person and audios where the

		Subtitling window size			
Evaluation	Type	2×163		5×200	
Questionnaires	audio	10	0.27 ±0.14	8	**0.31 ±0.14**
	talking	9	0.31 ±0.27	5	**0.39 ±0.22**
	video	5	0.22 ±0.12	3	**0.24 ±0.02**
	sum, avg	24	0.28 ±0.20	16	**0.32 ±0.17**
Continuous rating	audio	10	0.90 ±0.71	8	**1.66 ±0.95**
	talking	9	**1.65 ±0.52**	5	1.09 ±0.78
	video	5	1.11 ±0.50	3	**1.35 ±0.31**
	sum, avg	22	1.21 ±0.70	16	**1.42 ±0.85**

		Subtitling window size			
Evaluation	Type	18×250		5×200	
Questionnaires	audio	11	0.21 ±0.14	8	**0.31 ±0.14**
Continuous rating	audio	11	1.50 ±0.79	8	**1.66 ±0.95**

Table 4.4: Comprehension results of two experiments that compare settings by the history length. The upper part of the table addresses Small vs Medium experiment and the lower part Medium vs Large which is restricted to audio documents only. The sizes of the subtitling window are in the format (lines × millimeters) and the representation of the result numbers is the same as in Table 4.2.

illustrative or visually informative content is naturally not present. This is logical because users do not have to watch the video image, they only read the subtitles which have more reserved space for the history.

On the other hand, videos that provide information through the image do not appear to be notably better. It may be caused by possible complications when users switch their focus from the bottom part of subtitles to the video image.

Even though continuous rating reports preference on the smaller subtitling window size for videos of a talking person, the overall preference is better for the bigger window. Thus, we conclude that our hypothesis that more information is caught when using longer history is correct.

4.1.4 Audio: Medium vs Large

When comparing settings based on the subtitling window size, we extended the previous experiment to extremes. Recall the theoretical layout whose subtitling window height is set to 18 lines, which fits the video height having a default scale. Since the subtitling window is really large, we restricted the usage only for audios because it would not fit the screen otherwise. Our assumption is the following:

		Flicker Highlighting	
Evaluation	Type	No	Yes
Questionnaires	audio	14 0.24 ±0.15	13 **0.30 ±0.13**
Continuous rating	audio	14 1.32 ±0.82	13 **1.42 ±0.74**

Table 4.5: The level of user comprehension depending on whether the flicker state is highlighted and when it is not. The description of numbers as in Table 4.2.

Hypothesis 3.4.6 (Below Large & Comprehension)**.** *The Below Large layout achieves higher user comprehension compared to layouts that employ smaller subtitling windows.*

This benefit may be even more significant for audios because users do not have to watch the video image. They may return back in history as they wish, or read only stable parts of MT output.

The scores of comprehension for Below Medium and Below Large are captured in Table 4.4 (lower). The results show that the hypothesis is wrong — both evaluations report the increase in comprehension while using the Medium layout. Our explanation is that users can easily get confused and lost because of too many information present. Therefore, the conclusion is that the best layout is Below Medium when the comparison is based on a different size of the subtitling window.

4.1.5 Flicker Highlighting

Besides testing various sizes of the subtitling window, we can highlight the history graphically, namely its stable parts. In particular, we assign an appropriate colour to each flicker state and emit them as coloured background of the text. We expect that this approach is more appreciated because users are not surprised when the subtitles suddenly change. This way, users may follow these parts of subtitles which satisfy their comfort:

Hypothesis 3.4.11 (Flicker Highlighting & Comprehension)**.** *Flicker Highlighting aids in better understanding subtitles.*

The results are shown in Table 4.5. We can see that both evaluation approaches report the increase in comprehension when the flicker state highlighting is enabled. This confirms our assumption how beneficial the knowledge of the output stability distribution for the user is.

4.2 Readability

Although the evaluation of user comprehension is the most salient measure how to compare individual settings, there are other aspects which addresses user's satisfaction with translated speech subtitling. One of these factors is the ability of catching all subtitles, which we ask for at the end of each questionnaire where

Evaluation	Type	Below	Overlay
Readability	talking	9 2.67 ±1.05	9 **3.11** ±**0.87**
	video	5 2.20 ±0.75	8 **2.75** ±**0.83**
	sum, avg	14 2.50 ±0.98	17 **2.94** ±**0.87**

Table 4.6: Readability scores for each group of Below vs Overlay experiment on a discrete scale 1–5. The three numbers in each row and cell are the number of experiments, average and standard deviation. Higher score in each row bolded.

the participants may report a discrete score in the range between 1 and 5. For the sake of simplicity, we denote this evaluation shortly as *Readability*.

In the following, we discuss the same experiments as we analyzed in the previous section, but now from the perspective of readability. In particular, we compare the Below and Overlay layout, the Left and Below layout, different sizes of the subtitling window (2×163, 5×200 and 18×250), and examine the highlighting of the flicker state. In each experiment, we either confirm or deny hypotheses that we formulated in Section 3.4.

4.2.1 Overlay vs Below

The evaluation of Overlay vs Below experiment based on user satisfaction with reading subtitles is captured in Table 4.6. It can be seen that subtitles were read more easily when they were displayed over a video image. The results are, however, slightly unexpected given our assumption:

Hypothesis 3.4.2 (Below & Readability)**.** *Subtitles emitted below a video image are more readable than emitted over the image in the Overlay layout.*

We initially supposed that readability of white text on a black background, which is emitted over a video image, is worse when reading black text below the video with a white background. However, the participants reported the opposite, that is, they considered the subtitles to be more readable for all types of documents when the subtitles overlay the image.

We explain this preference by the fact that users are used to read subtitles over the image in movies, even though this composition is not naturally comfortable for eyes. On the other hand, we can see that the emission of subtitles over an illustrative content is less convenient than over videos of only taking people. This shows that our hypothesis is not entirely wrong — the background of videos of talking people is less visually disturbing for eyes and hence the readability is better, whereas videos with an illustrative content contain more image switching, which reflects on the reading discomfort.

4.2.2 Below vs Left

We now compare the Left layout to the Below Small layout from the perspective of readability, presenting the results in Table 4.7. When we evaluated user comprehension, the results of Left vs Below experiment were inconsistent — when the

Evaluation	Type	Left	Below
Readability	talking	5 **3.20 ±0.75**	7 2.71 ±1.16
	video	1 2.00 ±0.00	3 **2.67 ±0.47**
	sum, avg	6 **3.00 ±0.82**	10 2.70 ±1.00

Table 4.7: The results of contrasting the Left to Below layout, evaluated on readability. The score description is the same as in Table 4.6.

		Subtitling window size	
Evaluation	Type	2×163	5×200
Readability	audio	10 2.30 ±1.10	8 **3.62 ±1.22**
	talking	9 2.67 ±1.05	5 **3.40 ±1.20**
	video	5 **2.20 ±0.75**	3 1.67 ±0.47
	sum, avg	24 2.42 ±1.04	16 **3.19 ±1.33**

		Subtitling window size	
Evaluation	Type	18×250	5×200
Readability	audio	11 3.18 ±0.94	8 **3.62 ±1.22**

Table 4.8: The overview of result scores of readability for settings that solely differ in the subtitling window size. The upper part of the table compares Small to Medium, and the lower part Medium and Large. The latter is limited to audio documents only. The representation of numbers in each cell is the same as in Table 4.6.

Left layout was used, participants reported higher scores for videos of a talking person and lower scores for illustrative videos. We can see that the results of the reading preference match the results of user comprehension. However, note that the analysis of illustrative videos includes only one sample, which may indicate the benefit of the Below layout falsely.

Overall, the participants preferred the Left layout which was not expected because of very narrow subtitle lines. This rejects our hypothesis that better readability is acquired if the Below layout is used:

Hypothesis 3.4.10 (Left & Readability)**.** *The Left layout harms the quality of subtitle readability.*

Along with the collected scores of user comprehension the Left layout becomes generally more convenient. But we can use it only in situations when the space from the side of a video is available.

4.2.3 Video: Small vs Medium

Besides the position of the subtitling window, its size is probably the second important factor that influences user reading satisfaction while presenting subtitles. Table 4.8 (upper) captures gathered scores on readability for several settings, divided into three groups as usual.

We can see that the overall preference is in favour of the Below Middle layout, specifically when the subtitling window size is 5 lines long and 200 millimeters wide. There are also two essential observations about these results.

First, the most salient difference in the scores is for documents of the audio type only. It clearly shows that longer history is crucial for the ability of catching all subtitles. The explanation is that users have the opportunity to read the part of subtitles that is deeper in history and they do not lost the trail of reading. Note that this conclusion depends only on readability because audios do not contain any image which would influence the results.

Second, there is an inconsistency for illustrative videos where the participants consider subtitles in the smaller subtitling window more readable. It is probably caused by easier eye movement from a video image to subtitles and back because the subtitling window is shorter, i.e. eyes have to overcome less distance during focus switching. Recall that we exactly expected this preference when discussing theoretical layouts:

Hypothesis 3.4.4 (Below Small & Readability)**.** *Readability of subtitles in the Below Small layout is better compared to subtitling windows that contain more lines, especially for illustrative videos.*

We can conclude that the hypothesis is correct for videos with a visually informative content, but not for all types of documents as we originally assumed.

4.2.4 Audio: Medium vs Large

Similarly to the previous experiment, we also compare the Below Middle layout to the Below Large layout from the readability perspective. The results of this comparison can be seen in Table 4.8 (lower).

The results show the readability preference of the Below Middle layout, which matches the results of user comprehension. In both cases we hypothesized that the obtained scores would be higher for the larger subtitling window size, namely 18 lines high and 250 millimeters wide in the Below Large layout. For the sake of completeness, the readability hypothesis was:

Hypothesis 3.4.7 (Below Large & Readability)**.** *Below Large benefits from better readability compared to layouts of smaller subtitling windows.*

We can see that our assumptions about benefits of the Below Large layout are wrong. Despite the long history in the Below Large layout, users prefer the subtitling window to be shorter. It is probably caused by the undesired subtitling behaviour when a reset occurs. When this happens, all lines move up and having so long paragraph of subtitles, one can easily get lost which line he or she was focused on. Similarly, having less lines helps to spot keywords even when a reset occurs.

		Flicker Highlighting	
Evaluation	Type	No	Yes
Readability	audio	14 3.36 ±0.89	13 **3.77 ±0.89**

Table 4.9: Readability scores of the experiment which compares settings when the flicker state is highlighted and when it is not. The description of numbers is the same as in Table 4.2.

4.2.5 Flicker Highlighting

The last experiment and its analysis from the perspective of readability is Flicker Highlighting. We used the subtitling window of the size 18 lines and 250 millimeters, once with the flicker state highlighting and once without it. The resulted scores can be found in Table 4.9.

Our assumption was that the flicker state highlighting aids to comfortably reading subtitles because users can follow only stable parts of MT output:

Hypothesis 3.4.12 (Flicker Highlighting & Readability)**.** *Subtitles emitted with highlighted words regarding their flicker state are more readable.*

The results show that the assumption is correct — the scores for the setting with highlighting is higher. On the other hand, the improvement is not so significant, which can be explained by a very large subtitling window — the Below Large layout itself is not preferable.

4.3 Watching comfort

In the previous section, we discussed the preference of users based on their subjective perception of the reading quality — whether they managed to read all the subtitles (along with watching the video image if possible). This section is, on the other hand, related to the complementary question: When a document contains an image, how much does a user manage to watch it (along with reading the subtitles)?

Since videos of a talking person and audios do not contain informative content in the image, we have to exclude from the analysis such experiments (or their parts) which are related to one of these two document types. We discard the comparison of a subtitling window size in Medium vs Large experiment and Flicker Highlighting. Both these experiments use the audio document type.

Investigated experiments are thus consisted of the following three: Overlay vs Below, Below vs Left and Small vs Medium experiment. Within every of these three experiments we examine only videos with an illustrative content, which are relevant in this evaluation.

For the evaluation of all experiments in this section, we used scores from the end of each questionnaire where the participants reported a discrete value from the scale ranging between 1 and 5. For the sake of clarity, we denote this evaluation in the resulting tables shortly as *Watching*.

Evaluation	Type	Below	Overlay
Watching	video	5 2.20 ±1.60	8 **3.00 ±1.00**

Table 4.10: The overview of user satisfaction with watching comfort for Below vs Overlay experiment. The numbers are represented in the same way as in Table 4.6.

Evaluation	Type	Left	Below
Watching	video	1 2.00 ±0.00	3 **3.00 ±1.63**

Table 4.11: The results of the comparison of the Left and Below layout using the watching comfort evaluation. The score description is the same as in Table 4.6.

4.3.1 Overlay vs Below

As usual, we start with analyzing Below vs Overlay experiment. Note that we have no specific preference assumption based on watching comfort. Emitting subtitles below a video may help users to watch the video content comfortably because the image is not covered by any obstacles and can be fully informative. On the other hand, subtitles that are displayed over the image are closer to the center of the video and watching is not spoiled by long focus switching to the subtitles.

The scores of watching comfort are captured in Table 4.10. The results show that participants were able to watch subtitles more comfortably when using the Overlay layout rather than the Below layout. We explain this preference by casual watching movies with overlaid subtitles and the fact that the subtitling window size is not extremely large, i.e. it covers a small portion of the video image. Thus, watching is not spoiled by imperfect conditions of emitting subtitles over a video image.

4.3.2 Below vs Left

The next comparison is based on a different position of the subtitling window. Specifically, we examine the level of watching comfort in Left vs Below experiment, the scores of which are depicted in Table 4.11.

We can see that user preference inclines towards the Below layout even though we supposed that watching the image is more convenient in the Left layout — users continue with their look in the direction of reading where they spot the video image:

Hypothesis 3.4.9 (Left & Watching comfort)**.** *Watching a video image is more comfortable while using the Left layout rather than using the Below layout.*

On the other hand, the results may not be reliable because the sample size of setting instances is small. Since the readability and comprehension evaluation reported higher scores for the Left layout, having the situation where the video is not visually informative makes this layout advantageous.

		Subtitling window size	
Evaluation	Type	2×163	5×200
Watching	video	5 2.20 ±1.60	3 **2.33 ±1.25**

Table 4.12: The experiment comparing different sizes of the subtitling window. The used evaluation is watching comfort and the description of numbers is as in Table 4.6.

4.3.3 Video: Small vs Medium

The last experiment is devoted to the comparison of layouts which differ in the size of their subtitling windows. In particular, we examine user preference of the Below Small layout with 2 lines per 163 millimeters and the Below Medium layout with 5 lines per 200 millimeters. The resulting scores can be seen in Table 4.12.

The results report the inclination towards the subtitling window of the larger size, but the difference is not so significant. It seems that this larger size also prevent to watch the video image comfortably when reading bottom lines of subtitles. It is probably caused because eyes are forced to overcome a longer distance. Recall that we assumed the same:

Hypothesis 3.4.3 (Below Small & Watching comfort)**.** *The watching comfort of the Below Small layout is higher compared to layouts that employ larger subtitling windows, especially for illustrative videos.*

It is hard to decide whether the hypothesis is proven or not. But together with the evaluation of user comprehension and readability, we can say that the Below Medium layout is suitable in situations if enough space below the video is available. Especially, having more subtitle lines is beneficial for audios or for videos with a visually non-informative content.

4.4 Overall impression

Until now, we have compared settings in experiments that addressed some of our hypotheses. In the experiments, only these instances of settings were selected which had at least one common document in both compared settings. For example, if we examined layouts Below and Overlay, we chose only setting instances from the intersection of document subsets of layouts Below and Overlay.

However, we can analyze the layouts independently. For this evaluation, we used the score from the general part of questionnaires where we asked the following question: What was your overall impression of the subtitles?

The score was reported from a discrete scale from 1 to 5. The results are depicted in Figure 4.1. We can see that the most popular layout is when the flicker state is highlighted. We used this layout with audios and with the subtitling window size of 5 lines per 250 millimeters. Although previous analyses reported less satisfaction with readability, watching comfort and worse user comprehension in such a large subtitling window, overall impression of Flicker Highlighting beats other layouts. We can see that despite the fact that a large subtitling window is not convenient, the idea of the flicker state highlighting is clearly preferable.

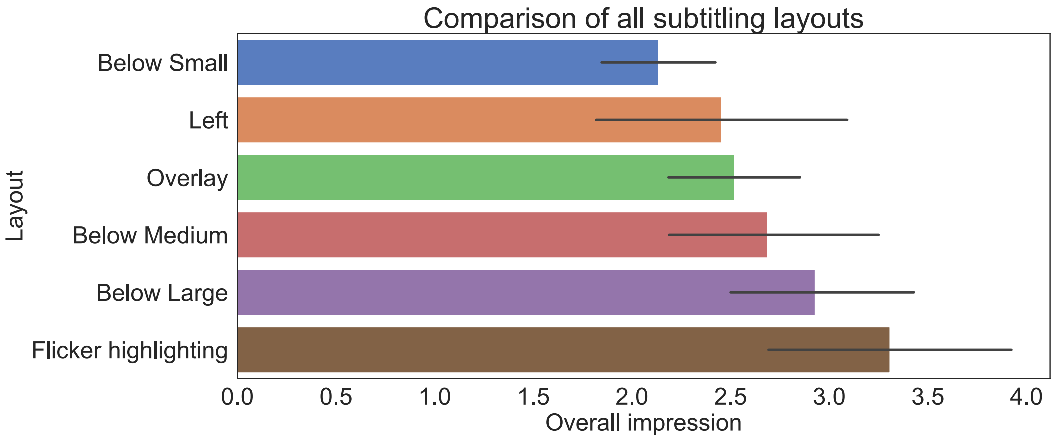


Figure 4.1: The scores of the overall impression which ware collected from the general part of questionnaires.

Thus, employing this idea also in other layouts may be a useful experiment as a future work.

Contrarily, positioning the subtitling window below a document with only two lines makes the Below Small layout the least favourite. We can see that participants see minimum advantages of emitting subtitles below a document when the history is short. In cases of longer history, Below Medium and Below Large are highly rated. On the other hand, when the usage of short history is necessary (e.g. because of spatial limitations), the Overlay layout is a better option in comparison to the Below Small layout. There is also a possibility to display subtitles on the left side of a video, which is not as popular as other layouts with longer history, but the comprehension and readability is reported to be sufficient.

4.5 Summary

This whole chapter has been devoted to the main goal of this thesis — to compare and analyze which layouts are more suitable for the use given spatial conditions or other requirements. Particularly, we modelled several settings from theoretical layouts which we then paired and contrasted in five experiments. The experiments were created according to our hypotheses whose goal was to show the preference of users based on four main evaluating techniques: Questionnaires, Readability, Watching comfort and Overall impression. Additionally, we used continuous rating as a secondary measurement which was used for the comparison to the evaluation of factual questions. We discuss the results of continuous rating in more detail in Section 5.2.2.

Here, at the end of this chapter, we summarize our findings and formulate recommendations when to use a particular layout given specific requirements. For the sake of simplicity, we do not report the numbers again, we only present a reduced comparison that only indicates which setting from each pair gets higher scores regarding used evaluation metrics. The comparison can be seen in Table 4.13.

			Evaluation			
Experiment	Layout	Description	Q	R	W	I
Overlay vs Below	A	Overlay	*	*	*	*
	B	Below				
Below vs Left	B	Below			*	
	E	Left	*	*		*
Small vs Medium	B	Small				
	C	Medium	*	*	*	*
Medium vs Large	C	Medium	*	*	-	
	D	Large			-	*
Flicker Highlighting	D	No			-	
	F	Yes	*	*	-	*

Table 4.13: The summary of all evaluation approaches that were used in our analysis. The meaning of observations is: Questionnaires (Q), Readability (R), Watching (W) and Overall impression (I). Stars denote better score in each pair of compared settings and a hyphen indicates that selected settings were not evaluated in given conditions.

We can see in the first experiment (Overlay vs Below) that emitting subtitles over a video image in the Overlay layout outperforms the Below layout in every evaluation. Remember that Overlay got especially high scores for readability when the document type was a video with a talking person — the subtitles did not interfere with the image informativeness. Altogether with results of user comprehension, watching comfort and overall impression, we recommend to use the Overlay layout when 2 lines not wider than the video are available.

The Left layout was tested in the second experiment (Left vs Below). Although the scores of comprehension are numerically better, the difference is insignificant. Videos of a talking person show an increase in comprehension when the history is longer, which is also confirmed by the results of (1) Watching comfort — users consider following the video image more uncomfortable when the video is visually informative — and (2) Readability — the subtitles are worse readable when the video image content is important. At the same time, videos of a talking person are suitable for the Left layout by all evaluations. Thus, our suggestion is to prefer the Left layout mainly if the video is not informative in the image.

The Below Small layout is the most unpopular layout in the Overall impression evaluation. The advantage of this layout was found in the Left vs Below experiment in the situation when illustrative videos were used, reported by Readability and Questionnaires evaluations. However, the results may be unreliable due to an insufficient sample size. Another increase in readability preference was shown in the experiment Below Small vs Below Medium, similarly in the case of visually informative videos.

Among all Below layouts, namely Below Small, Medium and Large, the highest total scores were observed when the subtitling size was 5 lines high and 200 millimeters wide. Below Middle outperformed its counterparts in three out of four evaluations — Questionnaires, Watching comfort and Readability. In the case of videos with an illustrative content the results are worse in comparison to Below Small for readability. Additionally, overall impression raises the preference of Below Large over Below Medium. Based on the results, we do recommend to use Below Medium, especially in situations when the document type is an audio or a video of a talking person.

Below Large obtained the highest scores in the Overall impression evaluation if we discard the results of Flicker Highlighting. On the other hand, comparing to Below Medium the usage of such a large subtitling window is reported to be worse in all other evaluations. Thus, we suggest not to use this layout.

The last studied layout was Flicker Highlighting. This layout is the most popular and together with other results of Questionnaires and Readability, which confirm better scores, the usage is highly recommended. Furthermore, we suggest to utilize flicker state highlighting in the combination with other layouts, supposing that it improves the quality of the subtitled presentation even more.

In conclusion, among all layouts we propose to employ Overlay, Left and Below Medium, possibly combining them with flicker state highlighting. The concrete selection depends on spatial limitations and a document type. On the other hand, we do not recommend to use Below Small and Below Large because of their history length, which resides both extremes.

5. General results and discussion

The subtitling layout analysis brought many observations, but several details of either the evaluation process or subtitling quality were omitted to keep the text simple and clear. In this chapter, we address the missing parts and discuss the results of the experiments from a more general point of view.

Section 5.1.1 is devoted to the analysis of subtitling quality. In particular, we first investigate user comprehension as subsequently removing top layers from our proposed structure of presentation hierarchy. Then, in Section 5.1.2 and 5.1.3, we address the quality of subtitles from the perspective of latency and fluency, respectively. We also examine both aspects in terms of developing an automatic evaluation metric as an alternative and faster way in parallel to user rating.

Section 5.2.1 describes the process of the questionnaire evaluation in more detail. In the following Section 5.2.2, we compare questionnaires and continuous rating with the intention of finding any dependence between them. The last part of this chapter, namely Section 5.2.3, is aimed at the relation between the information source of answers and continuous rating among German and non-German speaking group of participants.

5.1 Subtitling quality

Comprehension is definitely influenced by multiple factors, part of which we examined in previous chapter while empirically testing several subtitling layouts. The results indicated the preferred arrangement across pairs of settings, i.e which of two settings is relatively better. However, we basically obtained only a recommendation for the position and size of the subtitling window. Thus, it would be useful to measure the exact level of comprehension, and additionally answer following questions: What is the overall comprehension of an online setup? What portion of translation is lost solely because of flicker? How much substantial is the difference between an online and offline setup?

In Section 5.1.1, all aforementioned question are addressed and empirically answered. We show that there exists a subsequent loss in user comprehension as we move from the basic reading of offline machine translation to the more online and automatic speech processing. We also notice that user comprehension heavily depends on user attention and concentration.

In the next section, we demonstrate that subtitling quality is related not only to subtitling window size or position, but also to the quality of used translation system and its efficiency — especially for online subtitling, slow translating may produce delayed outputs which spoil the fluency of the presentation. Because we consider latency and fluency as two additional important aspects of presentation, we analyze their impact on user satisfaction in Section 5.1.2 and 5.1.3, respectively.

For the purpose of analyzing, during each experiment we collected all logs that were described in Section 2.1.4.

Type	w. avg±std	t-test
Offline+voting	0.81±0.11	
Offline	0.59±0.16	***
Online, without flicker	0.36±0.16	***
Online, flicker, top layout	0.33±0.13	
Online, flicker, least preferred	0.31±0.16	

Table 5.1: The scores of the hierarchical composition on all instances of settings. The average is weighted by the number of questions in document. Stars denote the statistically significant difference (p-value< 0.01) between the current and previous line.

5.1.1 Comprehension

Along with given spatial restrictions and a chosen layout, the user satisfaction with simultaneous translated speech subtitling heavily depends on the quality of a provided pipeline of language processing. The pipeline is traditionally composed of speech recognition, segmentation, machine translation and other intermediate procedures that join all together. Its output is then presented to users, employing one of subtitling layouts that we described.

Each of these components works unreliably, which typically leads to error accumulation. Thus, in this section, we split the whole system into several parts, after each we subsequently investigate the level of user comprehension. The comprehension is assumed to be assessed by a proportion of correctly answered questions, so the analysis uses questionnaires as our main evaluation metric.

Since breaking down the pipeline of language processing is beyond the scope of this thesis, we mainly focus on the second part, which is the presentation of subtitles from MT output. We assume that during answering the questionnaire, a person who perfectly understands the source language and is allowed to return back in the document arbitrarily can answer all questions correctly. Then:

- With a language barrier and offline MT (the user is allowed to seek through the document), some information may be lost in MT.
- More information is lost when only one perusal of the document is available during online MT. It is caused by forgetting and temporal inattention.
- Some more information may be lost because of flicker, i.e. it usually happens that subtitles that are already emitted have to be replaced because MT changes the confidence given a new output from ASR.
- Even more information may be lost because of a suboptimal subtitling layout, which we analyzed in the previous chapter.

Our findings confirm that we assumed the structure of comprehension levels correctly. Furthermore, we noticed that even the participants with unlimited access to MT output gave inconsistent answers, i.e. a question was answered correctly by one participant and wrongly by another. Thus, we combined the answers and marked the question correct if at least one of the answers was correct. This more relaxed variant of the score estimates the upper bound of accessible information. We explain the lower scores of a single person by participants' insufficient attention, although some unclarity of the questions or the text may be also the reason.

Table 5.1 summarizes the results on all documents. The results show that 81% of information was preserved by MT on average (Offline+voting, i.e. one of two participants answered correctly). A single participant was able to find 59% of the information (Offline).

We then constructed an artificial experiment without flicker where final outputs of MT were displayed at the time when only unstable was known (i.e. as if the system can always predict the best translation of the upcoming sentence). Investigating this experiment we found that a single participant was able to answer 36% questions correctly.

When the most preferred subtitling layout with flicker was used during online setup (Online, flicker, top layout), 33% of information was acquired. Having the least preferred layout but the same setup (Online, flicker, least preferred), information found dropped to 31%.

We found a statistical significance (two-sided t-test) between some pairs of experiments: Between offline MT with voting and without it, and between offline MT and online. The other differences are statistically insignificant.

5.1.2 Reading speed and latency

Another factor which influences the quality of presentation is the average difference between timestamps in which a word finalizes in translation (i.e. it no longer changes) and in subtitles (i.e. it is emitted to users). For every word, we denote this difference as a *latency* or *delay*.[1] One of the benefits of offline manual subtitling is the knowledge of time boundaries for each utterance in the source sound, which help to find a balance between the duration of utterance emission and the time (and shift) the corresponding subtitle is shown on screen. This is obviously not the case when a subtitle annotator is replaced by a system which automatically generates simultaneous translation as a speaker is speaking.

The problem of simultaneous speech subtitling is the fact that we do not know in advance what the next utterance is and how fast it will be spoken. For that reason, our Subtitler supports so-called adaptive reading speed (Section 2.1.2), which basically adjusts scrolling of subtitles according to the speed of the speaker's speech subject to a pre-defined minimum and maximum value.

To show that our proposed policy is more advantageous than fixed reading speed, we measure the delay of all presented words when the Below Small layout is employed. We use only Below Small in this experiment because it represents an upper bound for subtitling delay of bigger subtitling windows.

First of all, we need to find an appropriate value for fixed reading speed. It can be derived from the recommendations as shown in Section 2.1.3. However,

[1] Note that our defined term *latency* considers only the time when a word is translated, not spoken. The reason is that we do not have time alignments of uttered words in the source speech. However, this makes the latency shifted only by a constant.

	Delay						
	70%	80%	90%	95%	99%	max	resets
Adaptive r. speed	**0.01**	**1.44**	**3.06**	**4.51**	**7.05**	**12.06**	8.80
R. speed 18 char/sec	1.74	3.54	5.18	7.52	10.65	16.78	**5.47**

Table 5.2: The adaptive reading speed in comparison to the manually configured. Percentages denote the proportion of words that have a delay less than the given number. The delay is in seconds, resets in the average count per document.

we obtain its value more precisely by first running the experiment with adaptive reading speed and then computing the weighted average of logged values from each Subtitler update. The average results in 18.30, which we round to 18 characters per second.

Having computed the value of fixed reading speed, we run the experiment once with fixed and once with adaptive reading speed. For both passes, delays for every word are recorded, sorted and split by several milestones. The results are captured in Table 5.2.

We can see that the distribution of delays when fixed reading speed is used is inferior to adaptive reading speed. In particular, for manually set reading speed, 90% of all words in test documents are displayed in subtitles at most 5 seconds after translation, whereas adaptive reading speed ensures 90% of words to have the delay under 3 seconds. Similarly, in 99% of cases a word is emitted below 10.5 seconds after translation for fixed and below 7 seconds for adaptive reading speed.

The main benefit of the continuous adaptation of reading speed is the ability to quickly recover from reset occurrences. On the other hand, notice that it may spoil the reading flow because the number of resets increases. It can be seen in the last column of Table 5.2 that as the speed adaptation reveals 8.80 resets per document, the fixed speed keeps them at 5.47 per document. It is simply a direct drawback of adaptive reading speed — more up-to-date subtitles bring more subtitles that are not finished but already vanished in the history. Their updates then generate more resets. In other words, trying to stay more simultaneous is possible but with the current underlying retranslation MT system, this leads to less convenient behaviour. If the underlying MT system is stabilized more, avoiding unnecessary changes in older output, adaptive reading speed would not bring this negative effect.

Since adaptive reading speed outperforms its counterpart even though the value of fixed reading speed was computed on a very small sample and a specific domain, we definitely recommend to use adaptive reading speed unless low reset count is preferred. It is especially useful in situations when the presentation is heterogeneous, i.e. it contains multiple speakers who differ in speech speed.

We can also evaluate latency from the perspective of user perception. For this purpose, the general part of questionnaires contains a question which specifically addresses the participants' satisfaction with the level of subtitling delay: How do you rate the subtitles in terms of synchronization with the source?

However, contrasting instances of all settings would be too complicated for

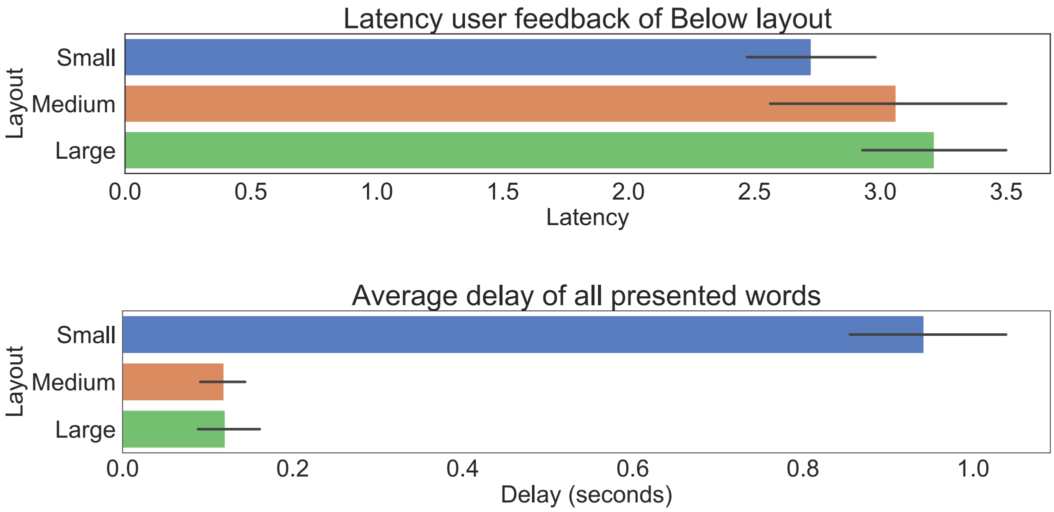


Figure 5.1: The user rating on latency (upper) and averaged delay across all words in one instance of setting (lower), as comparing three variants of Below layout.

making conclusions because the evaluation may be influenced by additional aspects, e.g. readability or watching comfort for different positioning of subtitling window. We thus compare layouts that share the position of their subtitling window, but differ solely in its size. A perfect candidate for this evaluation is the Below layout with its three variants — Small, Medium and Large.

Figure 5.1 (upper) shows the relation between aforementioned layouts as rated by participants' subjective assessment in a discrete scale from 1 (subtitles synchronized with the source the worst) to 5 (the best). We used 15, 15 and 5 (only audios) documents for the Small, Medium and Large layout, respectively. We can see that the results matches an intuitive assumption of layout ordering — the highest score is assigned to Below Large whereas the least delayed subtitles are considered to be in Below Small.

Having user perception of latency, it would be useful to derive a formula from observed logs (delays) and compare the computed value to the gathered scores from users. Hopefully, we obtain an automatic evaluation on latency and save time by omitting human evaluation.

Since our collected data contain delays of each word, we measure the average of delays of all words for one setting instance. Afterwards, the averaged delay is grouped by the layout and the results are depicted in Figure 5.1 (lower).

The most salient observation is that there is a big gap in computed delay between the variant Small and Medium, whereas the average delay for Medium and Large variant is very similar, almost the same. This simply implies that having at least 5 lines of subtitles in the Medium layout is sufficient to keep the average delay below 0.2 seconds. If we had a denser distribution, e.g. subtitling windows of lengths ranging from 2 to 5 lines, we would probably see smoother transitions between adjacent samples.

Nevertheless, computing the average of word delays is not a reliable metric how to automatically evaluate user perception of latency. Comparing participants' feedback from the general part of questionnaires to the true delay do not show

	German>=A2		German<A2		
flicker	3	**0.59±0.15**	10	0.30±0.15	sign. $p < 0.05$
no flicker	4	0.40±0.06	10	**0.34±0.07**	insignificant
t-test	$p < 0.10522$		insignificant		

Table 5.3: Comprehension scores on two documents on a setting with and without flicker, as rated by participants whose German language proficiency is between A2 and B2 on CEFR scale (elementary to upper intermediate), or below A2 (zero or beginner). The numbers in each cell denote the number of samples, average and standard deviation.

any observable relation. More sophisticated measurement is thus left as a future work.

5.1.3 Flicker and fluency

As we saw at the beginning of the previous section, latency is closely related to fluency — lower latency typically means lower fluency. By the term *fluency* we understand the level of subtitle disruptions, unpredictable updates or irregular resets.[2] Since the definition is very broad, we do not have any established method for its evaluation.

However, we assume that user behaviour varies as their knowledge of the source language differs. Specifically, the users who do not understand the source sound at all (or very minimally) focus on reading all subtitles all the time and do not pay much attention to the speech. Their key requirement is high quality translation with fluent emission of subtitles. Contrarily, having a presentation with low latency is not demanded.

On the other hand, the strategy of the users with a limited but nonzero knowledge of the source language is different. They listen to the speech and follow current subtitles simultaneously, as they try to get the best experience from the whole presentation (subtitles + sound + image). They follow subtitles when they are temporarily uncertain or need assistance with an unfamiliar word. It also means that they accept low latency at the expense of low fluency, and do not mind slightly lower translation quality.

To empirically test our assumption, we use the Below Small layout as a basis for our two contrasting settings:

- with flicker, that is, new incoming sentences are shown as soon as possible while obeying reading speed — worse fluency, and
- with no flicker, that is, all displayed sentences must be completed in order to be considered for presentation — better fluency.

[2] In the field of machine translation, *fluency* typically means a natural flow of words in a given language. We define *fluency* as a subtitling flow which is nevertheless proportional to its traditional meaning — more flicker means more unstable subtitles, which reflects on worse word orderliness.

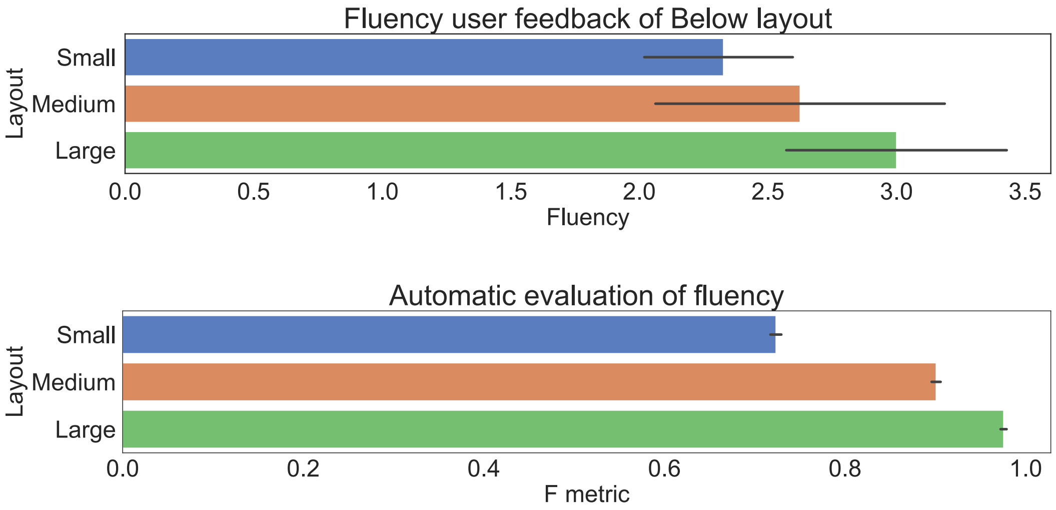


Figure 5.2: The scores of user rating on fluency (upper) and the proportion of unchanged words in each Subtitler update (lower).

In other words, the former forces the subtitles to be emitted immediately (as they are available) but with frequent rewriting, which may spoil users' reading comfort. The latter presents only final translations without rewriting but with a long latency. Note that the flicker parameter distinguishing the settings is configurable in Subtitler, which we described in Section 2.1.2.

We selected two videos and distributed these settings uniformly between German speaking and non-German speaking participants. We used questionnaires as our main evaluation metric and the distributions were analyzed by two-sided t-test.

The results are captured in Table 5.3. We can see that German-speaking users achieved higher comprehension with flicker than without, whereas the non-German speakers understood better without flicker. For the German group, the difference is very close to be statistically significant (p-value < 0.10522), but the difference for the non-German group is insignificant. The results of continuous rating confirm the trend of comprehension but the differences are insignificant.

We explain the insignificance of the results by the insufficient sample size of participants. We thus propose to extend this experiment by more users as a possible future work.

Similarly as we analyzed latency, fluency can be evaluated via user feedback. We asked participants to score the quality of fluency in the following question: How do you rate the subtitles in terms of fluency or language accuracy?

For the evaluation, we used again the Below layout and compared its three variants — Small, Medium and Large. We expect that smaller the subtitling window is, lower user satisfaction with fluency is. It is caused by insufficient history, which cannot be exploited if a user misses the translation because of flickering.

The scores can be seen in Figure 5.2 (upper), as rated by participants within a discrete scale from 1 (the worst fluency) to 5 (the best). The distribution of documents is the same as in the latency analysis, i.e. 15, 15, 5 for the Small, Medium and Large layout, respectively.

The results show that the least fluent layout is Below Small and the highest score was achieved when Below Large is used, which is according to our expectations. More questionable is whether we can find any automatic evaluation based on collected logs which would match the scores of the manual evaluation.

We have two types of logs which we can benefit from — the reset count and the number of removed words in each Subtitler update (see Section 2.1.2). The former is not usable in our situation because Medium and Large layouts contain too large subtitling windows that have zero reset occurrences. On the other hand, the latter seems to be promising, but we need to process it first.

We model it by following: Since bigger subtitling windows contain more stable subtitles, instead of taking raw counts of word deletions we consider the ratio between the counts and all words displayed during the given update. Afterwards, we compute the average of these skewed counts across all Subtitler updates. For the sake of simplicity, we subtract the average from one to have the metric proportional to the user feedback. Formally, if we denote N_i and R_i to be the number of displayed words and the number of removed words in the update i of all updates I, respectively, then the automatic fluency metric F is computed as:

$$F = 1 - \frac{1}{I}\sum_{i=1}^{I}\frac{R_i}{N_i}.$$

The results of the fluency error rate divided by the layout are plotted in Figure 5.2 (lower). For the Below Large layout, during each update of Subtitler the number of updated words is very low in comparison to all displayed words ($F = 0.9756$). On the other hand, almost 30% of words in each update are deleted and replaced ($F = 0.7235$) when using Below Small. The Below Medium layout achieves $F = 0.9008$.

Since our proposed metric returns the same order of layouts as participants rate, we suggest to use this metric as an estimate of a relative quality between two layouts, i.e. to answer the question which layout is better/worse in terms of subtitling fluency. This way, the process of evaluation can become more efficient, discarding one question in user inquiry.

5.2 Questionnaires

Questionnaires were primarily introduced as a tool of the subtitle quality estimation as finding the level of user comprehension. We also saw that they can be used as a complementary inquiry, which addresses general aspects of the user satisfaction with either the whole presentation (readability, watching comfort) or with the subtitling quality (latency, fluency).

Although the evaluation of general part of questionnaires is straightforward and does not require additional processing, a proper analysis of answers of factual questions is necessary. For this purpose, we sketched the usage of weights in Section 3.1.2 but omitted the analysis. Here, in Section 5.2.1, we complete the analysis by showing the selection of weights in more detail.

Questionnaires were our main evaluation metric but note that we also employed continuous rating as our secondary technique how to estimate user comprehension. In Section 5.2.2, we examine the relation between questionnaires and

Label	Type	OK	OK-	FG	UKN	WR
$\geq$ OK	positively-oriented	1.0	-	-	-	-
$\geq$ OK$-$		1.0	1.0	-	-	-
BASE	neutrally-oriented	1.0	0.5	0.1	-	-
$\leq$ UKN	negatively-oriented	-	-	-	0.1	1.0

Table 5.4: Possible approaches how to assign weights to each category. For sake of clarity, zero weights are denoted by hyphen.

continuous rating. Knowing any dependence among them would possibly allow replacing the questionnaires by continuous rating and save the time of tedious questionnaire creation and its later evaluation.

Finally, in Section 5.2.3, we investigate the distribution of information source of answers, the data of which we collected by a complementary question after each factual question. The analysis is performed on German and non-German speaking participants separately. We show that the distinction between these groups is significant, which proves that the group selection was done correctly.

5.2.1 Selection of weights

In Section 3.1.2, we described our requirements on the creation of questionnaires and presented a way how to evaluate responses of factual questions. We introduced so-called *weights* whose main purpose is to determine how much we trust each answer. More precisely, we first classify each answer into one of five categories based on the answer key: correct (OK), partially correct (OK-), wrong (WR), unknown (UNK) and forgotten (FG). Then, a decimal weight within the range from zero to one is assigned to every category. Finally, the total score of the observed setting instance is obtained by computing a weighted average through all its categories.

Now, we define three basic types of evaluation *weightings*: positively-oriented, neutrally-oriented and negatively-oriented, where the term weighting refers to the tuple of weights. A positive weighting requires its FG, UKN and WR groups to be set to zero. On the contrary, OK and OK- groups are zeroed as a requirement of a negative weighting. A neutral weighting may have the categories valued arbitrarily.

We suggest several approaches how to value weightings, which are captured in Table 5.4. The first row denotes weights where OK is equal to one and the rest to zero. The second row describes a setup where OK and OK- are set to one, and the rest to zero. Both these weighting systems are *positively*-oriented since we trust correct answers the most and do not penalize for wrong answers.

The third row captures a *neutrally*-oriented weighting: OK = 1.0, OK- = 0.5, FG = 0.1 and the rest equals to zero. We can see that the weight that refers to partial correctness is set to 0.5, which makes the weighting more balanced in comparison to the evaluation of fully correct answers. We also raise the weight of forgetting because even though its true evaluation may be wrong (some participants may unconsciously want to boost their score or they really saw the answer

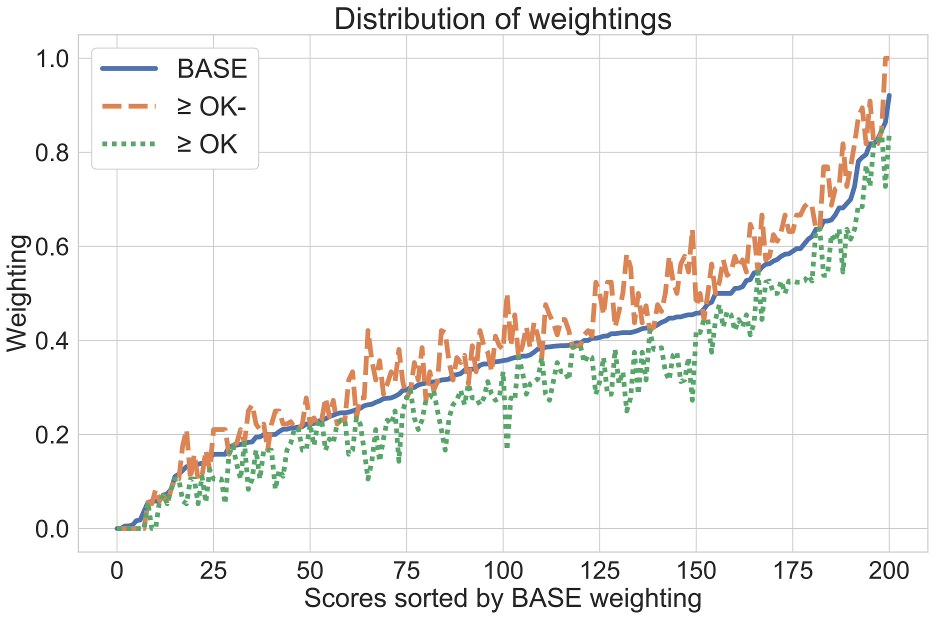


Figure 5.3: The comparison of positively-oriented weightings ≥ OK−, ≥ OK (upper) and a negatively-oriented weighting ≤ UKN (lower) to BASE.

but the translation was wrong), there are definitely some cases when the answer would be really correct if they could have remembered it.

Furthermore, we may be interested in negative feedback as well, i.e. to get a higher score when the answer is wrong. In particular, the forth row describes *negatively*-oriented weights: WG = 1.0, UKN = 0.1 and the rest is set to zero. This approach is helpful in situations when we compare several translation systems, searching for the parts of MT output which are often mistranslated. Similarly, this weighting can be used for filtering out participants who do not take the task seriously, do not pay attention during watching and answer irrelevant responds.

However, these weighting types are constructed only theoretically. Therefore, we need to examine how each weighting behaves in practice. Specifically, we investigate whether they are entirely different or if we can find some similarities. We then pick one which satisfies our requirements and is suitable for the use in analyses.

The comparison can be seen in Figure 5.3. We sorted all setting instances by the resulting score (which we computed as weighted average) of the neutrally-oriented weighting (BASE), and plotted three scores (BASE, ≥ OK, ≥ OK−). We can see that the trend is very similar for all weightings and the baseline weighting behaves like the average of both positively-oriented weightings. The reason is that correct answers are salient in the comparison because all weightings assign the highest possible weight to this category. Fluctuations of ≥ OK and ≥ OK− weightings are caused by overestimating or underestimating the OK-category.

Since the comparison shows that the baseline smooths the score the best and the meaning of assigned weights corresponds to the behaviour of participants in reality, we decided to use this weighting for the questionnaire evaluation in our subtitling layout experiments. Nevertheless, other weightings may be also useful for different applications, several of which we already mentioned.

5.2.2 Relation to continuous rating

We presented questionnaires as a new approach how to evaluate user comprehension during subtitling a simultaneous speech. We also used continuous rating as our secondary measure where users have an opportunity to express their satisfaction with the quality of subtitles at given times within four levels, ranging between 0 (the worst) and 3 (the best). Additionally, for each question from questionnaires we know the time range when the necessary piece of information is uttered in the source speech document.

Based on this timing information, we can examine the relation between the proportion of correctly answered questions from questionnaires and the reported continuous feedback. Figure 5.4 captures how many times participants pressed each button of continuous rating, divided by the category of evaluated answers from questionnaires. Specifically, we merged correct (OK) and partially correct (OK-) answers to one category (OK/OK-), which represented parts of documents that were understood acceptably. The remaining categories sustained unchanged as we labeled them according to the answer key, i.e. spotted but forgotten (FG), missed by the user (UKN) or misunderstood (WR). This data aggregates observations for all instances of settings that solely depend on the subtitling layout change (discussed in Section 3.4), i.e. we excluded the offline machine translation and the artificial online machine translation without flicker.

Our goal was to find any dependence between the distribution of answer evaluation and continuous rating. We measured this relation by χ^2-test, and the results of the test are shown in Table 5.5.

Although the data for the non-German speaking group reports no dependence between these two distributions, the German speaking participants are found to be more reliable in continuous feedback. We can see that there exists a statistically significant dependence between unknown answers and continuous rating and acceptable answers and the rating. We can thus predict predict their comprehension with a higher precision, if we know the score of continuous rating. On the other hand, forgetting and wrongly answered questions are independent on the continuous feedback.

An exemplary usage of these observations may be when two translation systems need to be compared. Participants who understand the source speech with elementary to upper intermediate skills are more suitable to only watch the subtitled presentation and press buttons of continuous rating. On the other hand, participants with zero knowledge or very limited skip the continuous rating and only fill the questionnaire.

The dependence among the German speaking group can be explained by their ability to include the adequacy factor into continuous rating. On contrary, the participants of non-German speaking group have no competence for rating the adequacy. Their comprehension is thus probably independent on the continuous feedback. The feedback is solely based on the combination of fluency, readability and flicker.

We show that there exist a dependence between both distributions, differing

		χ^2-test p-values			
Label	Type	Non-German speaking		German speaking	
WR	wrong	0.53	insignificant	0.81	insignificant
UKN	unknown	0.28	insignificant	**0.09**	sign. $p < 0.10$
FG	forgotten	0.69	insignificant	0.61	insignificant
OK/OK-	acceptable	0.12	insignificant	**0.03**	sign. $p < 0.05$

Table 5.5: χ^2-test for the statistical significance of the relation between two distributions: the answer correctness and continuous rating.

by the level of source language proficiency. This can be used in future works as a basis for less time-consuming evaluation without laborious preparation, answering and evaluation of questionnaires, if our participants have at least elementary knowledge of the source language.

5.2.3 Information source

When evaluating correctness of participants' responses on factual questions from questionnaires, the most salient finding is whether the answer is acceptable by the answer key. If it is, the presented information is successfully transmitted to the user and the score of the used setting increases.

Additional important output of the evaluation is the knowledge where the source of this information comes from. For that reason, we ask participants to answer a complementary inquiry after every factual question. The survey investigates whether the information is obtained from subtitles, a video image, sound or whether the answer had been known before the experiment began. In Figure 5.5, we can see the distribution of participants' clicks of continuous rating, divided by the information source of the corresponding answer from questionnaires. The data contain all setting instances except the offline machine translation and the artificial online machine translation without flicker. We can see three major observations.

First, the most frequently reported source of information is *not given*, i.e. participants skipped answering this complementary question. The reason is simple — this inquiry is not mandatory because the participants are expected to tick the option only if they fill in the answer of the corresponding factual question. It can be seen that the distribution of *not given* matches the pattern of *unknown* responses from Figure 5.4.

Second, more important, the German speaking group of participants reported that the information of some answers was found in the video image or in the sound, which indicates that this group was constructed correctly — they seek the source of information not only in subtitles but also in other modalities of the presentation.

Third, having the previous observation, it is important to notice that the non-German speaking participants followed subtitles as their main information source (excluding *not given*), suppressing the video image and the sound. The

	χ^2-test p-values			
Source	Non-German speaking		German speaking	
image & sound	**2.26e−3**	sign. $p < 0.01$	**2.95e−7**	sign. $p < 0.01$

Table 5.6: χ^2-test for the statistical significance of the relation between two distributions: the information source and continuous rating.

results also show that the factual questions were constructed correctly in terms of new information bringing to a user, as *known before* option is kept low (note that having zero is unreachable).

In order to prove our hypothesis that the participants are split into groups based on their language proficiency level appropriately, we perform χ^2-test to measure whether the distribution of the information source and continuous rating are independent or not. The result of the test is shown in Table 5.6.

For both groups, we obtained a statistically significant dependence between the information source and continuous rating. It means that the information gained from the image and sound is dependent on the continuous rating feedback and follows presented distribution. In the case of the non-German speaking group, the distribution resembles a uniform distribution, which implies that users of minimal source language knowledge equally badly exploit the information from the sound or image. On the other hand, among the German speaking group, the source of caught information is more likely to be correctly predicted based on the continuous rating feedback.

Having our assumption proven, it makes sense to differentiate users according to their knowledge of the source language. The user behaviour differs between groups, which may reflect on their preference — in Section 5.1.3, we saw that German speaking participants required less delay at the expense of more flicker, whereas the non-German speaking group preferred the opposite. In a future work, we suggest to analyze subtitling layouts from the perspective of the language preference if a larger sample of participants is available.

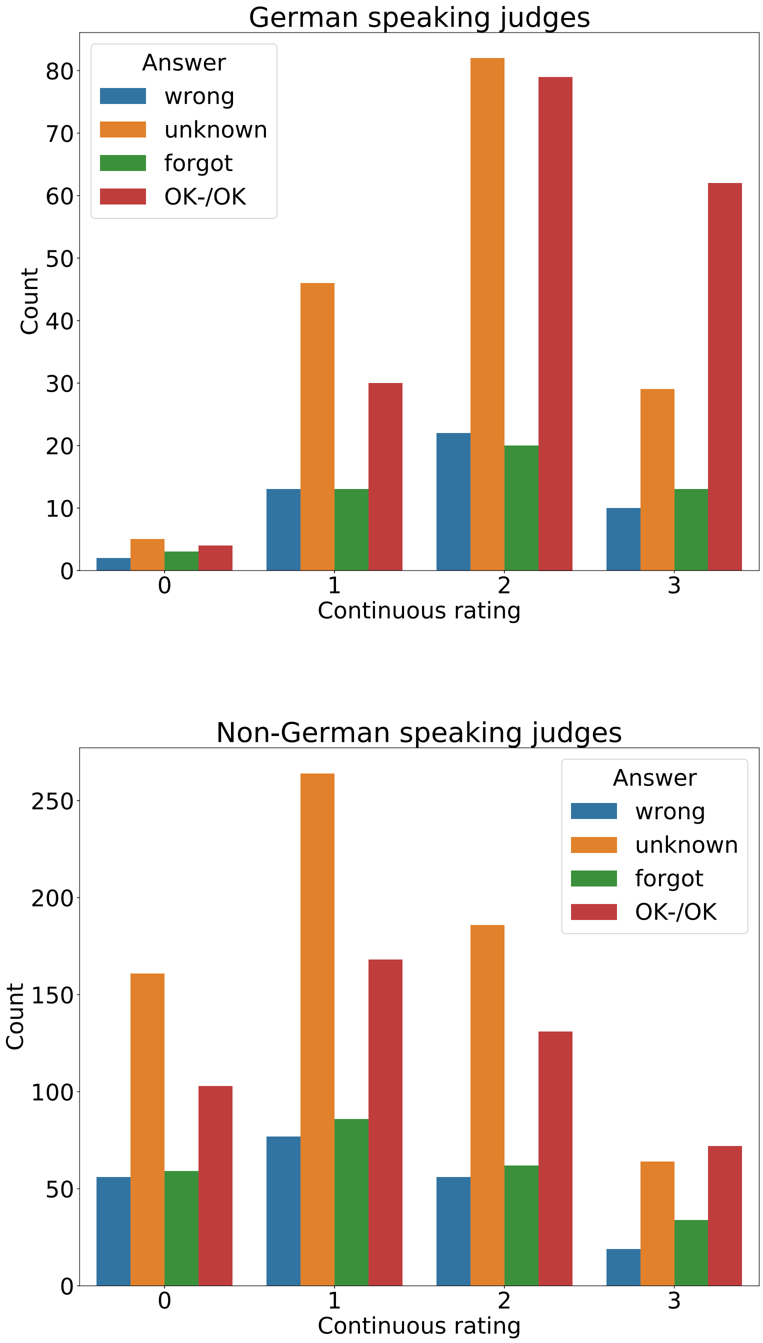


Figure 5.4: The distribution of the continuous rating and results of answers for non-German (upper) and German speaking (lower) participants.

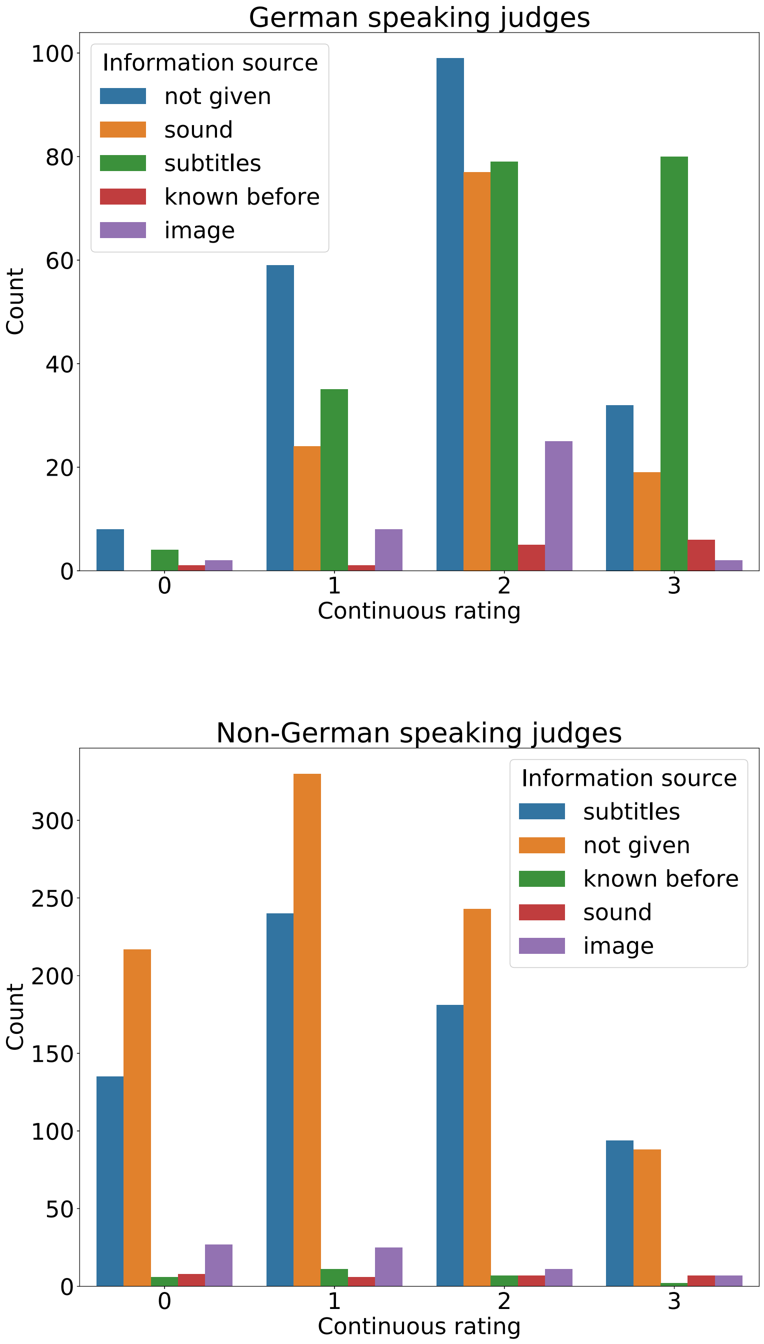


Figure 5.5: The distribution of the continuous rating and source of answers for non-German (upper) and German speaking (lower) participants.

Conclusion

This thesis empirically investigates viewer satisfaction with the presentation of simultaneous SLT subtitling. The experiments were based on the comparison of several subtitling layouts that were created according to hypotheses regarding comprehension, readability and watching comfort. The assumptions were formally formulated, empirically tested and analyzed.

The analysis was conducted as a result of two evaluation methods that we employed. First, continuous rating aimed at regular feedback from participants during the whole subtitling. It expressed current viewer satisfaction with the quality of presentation. Second, questionnaires were used as a more objective measurement of viewer comprehension, extended by an extra inquiry that addressed additional aspects of the presentation.

The experiments were performed using our proposed subtitling library called Subtitler, and using a presenting application that was created to simulate online SLT subtitling. The implementation of the application aimed at simplicity, clarity, reliability and robustness.

The results showed that the most preferred layout is Below Middle, which outperformed its counterparts Below Small and Below Large in three out of four evaluations — Questionnaires, Watching comfort and Readability. We also observed the increase in comprehension when Left and Overlay layouts were used, which are both suitable for deployment in situations when a free space below a video is not available. On the other hand, some layouts were found to be too disadvantageous and they can be excluded from future experiments, e.g. Below Small.

Our Subtitler also brought adaptive reading speed, a new feature which was not implemented in its predecessors. The analysis showed that adaptive reading speed ensured fast recovery from reset occurrences. On the other hand, it may spoil the reading flow because of the increased number of resets. Additionally, the fluency of presentation (as referring to the level of subtitle disruptions, unpredictable updates or irregular resets) was shown to be automatically derivable from the collected logs.

The examination of results from the general point of view led us to the conclusion that there exists a hierarchy of comprehension levels. In particular, we documented that 81% of information was accessible in offline MT to at least one of two participants. 59% of information was correctly identified by a single participant reading offline MT output. When presented as online subtitling without flickering, this estimate of comprehension sank to 36% and the usage of the least preferred layout decreased it to 31%.

We also showed that the preference of viewers depends on their knowledge of the source language. Participants who understood the source speech at least partially achieved higher comprehension with flicker than without, whereas participants with zero knowledge of the source language understood better if flicker was avoided by extending the delay. Additionally, dividing participants into two groups by their language proficiency is advantageous for future work, achieving a less time-consuming evaluation: The less proficient group is better fit for only watching the presentation and providing continuous rating. The more proficient group should skip the continuous rating and only fill the questionnaire, thereby saving time but delivering more reliable answers.

The tools that we created (esp. Subtitler but also the presenting application for running similar experiments as the one of ours), and collected responses with logs and their evaluation are attached to this thesis. The experiment and its analysis were submitted to ACL-IJCNLP 2021 conference and while the reviews were generally positive about the method and understudied and relevant topic, the paper was not accepted. For acceptance at this high-profile venue, a larger scale of the experiment will be necessary. We plan on adding new participants and when our paper eventually gets accepted, the full underlying dataset will be publicly published.

Future work

Possible directions of future research and implementation extensions were revealed during the work on this thesis:

- To obtain significant results it is necessary to extend the sample of participants and focus only on experiments that seem to be the most promising — we propose to aim at the Below Middle layout and the combination of Flicker Highlighting with other layouts.

- We found that dividing participants to two groups is logical since German speaking participants sought the information also in the image or sound, whereas the non-German speaking followed only subtitles. Having more participants may aid to a more detailed analysis of the layout preference based on the source language knowledge.

- Some information is definitely lost due to viewers' insufficient attention or memory. A possible way how to measure this loss is to compare the presentation (1) which employs MT for the generation of subtitles and (2) which uses transcribed translation from an interpreter as a replacement of MT output.

- Continuous rating may be an efficient way how to evaluate quality of several translation systems, if we employ participants who understand the source speech with elementary to upper intermediate skills. This provides a less time-consuming approach compared to questionnaires.

- Although we derived some automatic metrics from the collected logs for individual presentation properties, a possible extension is to propose a way how to automatically evaluate the whole simultaneous SLT subtitling. Such a metric would complement the standard translation quality, delay and flicker with some estimate of cognitive load.

Bibliography

Anthony. Why you should never use pure black for text or backgrounds, 2020. URL https://uxmovement.com/content/why-you-should-never-use-pure-black-for-text-or-backgrounds. Accessed: 2021-05-27.

Simon Arnfield, Peter Roach, Jane Setter, Peter Greasley, and Dave Horton. Emotional stress and speech tempo variation. In *Speech under Stress*, 1995.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

D Bauer and CR Cavonius. Improving the legibility of visual display units through contrast reversal. *Ergonomic aspects of visual display terminals*, pages 137–142, 1980.

BBC. BBC Subtitle Guidelines, Apr 2019. URL https://bbc.github.io/subtitle-guidelines. BBC © 2018 Version 1.1.8.

Jan Berka, Ondrej Bojar, et al. Quiz-based evaluation of machine translation. *The Prague Bulletin of Mathematical Linguistics*, 95:77, 2011.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation*, pages 12–58, 2014.

Peter F Brown, Stephen A Della Pietra, Vincent J Della Pietra, and Robert L Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311, 1993.

Lindsay Bywood, Panayota Georgakopoulou, and Thierry Etchegoyhen. Embracing the threat: machine translation as a solution for subtitling. *Perspectives*, 25(3):492–508, 2017.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. Re-evaluating the role of bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, 2006.

Chris Callison-Burch, Philipp Koehn, Christof Monz, and Josh Schroeder. Findings of the 2009 Workshop on Statistical Machine Translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 1–28, Athens, Greece, March 2009. Association for Computational Linguistics.

David Chiang. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05)*, pages 263–270, 2005.

Eunah Cho, Jan Niehues, and Alex Waibel. Segmentation and punctuation prediction in speech language translation using a monolingual translation system. In *International Workshop on Spoken Language Translation (IWSLT) 2012*, 2012.

Eunah Cho, Christian Fügen, Teresa Herrmann, Kevin Kilgour, Mohammed Mediani, Christian Mohr, Jan Niehues, Kay Rottmann, Christian Saam, Sebastian Stüker, et al. A real-world system for simultaneous translation of german lectures. In *INTERSPEECH*, pages 3473–3477, 2013.

Deborah Coughlin. Correlating automated and human assessments of machine translation quality. In *Proceedings of MT summit IX*, pages 63–70. Citeseer, 2003.

George Doddington. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, pages 138–145, 2002.

Yvette Graham, Timothy Baldwin, Alistair Moffat, and Justin Zobel. Continuous measurement scales in human evaluation of machine translation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 33–41, 2013.

Yvette Graham, Timothy Baldwin, and Nitika Mathur. Accurate evaluation of segment-level machine translation metrics. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1183–1191, 2015.

Rodrigo Laiola Guimarães, Jessica Oliveira Brito, and Celso AS Santos. Investigating the influence of subtitles synchronization in the viewer's quality of experience. In *Proceedings of the 17th Brazilian Symposium on Human Factors in Computing Systems*, pages 1–10, 2018.

Alina Karakanta, Matteo Negri, and Marco Turchi. Is 42 the answer to everything in subtitling-oriented speech translation? *arXiv preprint arXiv:2006.01080*, 2020.

Fotios Karamitroglou. A proposed set of subtitling standards in europe. *Translation journal*, 1998.

Philipp Koehn, Franz J Och, and Daniel Marcu. Statistical phrase-based translation. Technical report, University of Southern California Information Sciences Institute, 2003.

Maarit Koponen, Umut Sulubacak, Kaisa Vitikainen, and Jörg Tiedemann. Mt for subtitling: User evaluation of post-editing productivity. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 115–124, 2020.

Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612, 2004.

Dominik Macháček and Ondřej Bojar. Presenting simultaenous translation in limited space. In *Proceedings of the 20th Conference ITAT 2020: Workshop on Automata, Formal and Natural Languages (WAFNL 2020)*, 2020. In print.

Evgeny Matusov, Patrick Wilken, and Yota Georgakopoulou. Customizing neural machine translation for subtitling. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 82–93, 2019.

Rx Optical. Is dark mode better for your eyes?, 2020. URL `https://rxoptical.com/eye-health/is-dark-mode-better-for-your-eyes`. Accessed: 2021-05-27.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

François Pellegrino, Christophe Coupé, and Egidio Marsico. A cross-language perspective on speech information rate. *Language*, pages 539–558, 2011.

Yvette Shen. Line-height, 2012. URL `http://smad.jmu.edu/shen/webtype/lineheight.html`. Accessed: 2021-05-27.

Smrž. Online text flow library. `https://github.com/ELITR/online-text-flow`, 2020. Accessed: 2020-08-16.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Gareth Ford Williams. bbc. co. uk. online subtitling editorial guidelines v1. 1, 2009.

Philip Williams, Rico Sennrich, Matt Post, and Philipp Koehn. Syntax-based statistical machine translation. *Synthesis Lectures on Human Language Technologies*, 9(4):1–208, 2016.

List of Figures

List of Tables