SIMULTANEOUS SPEECH TO SPEECH TRANSLATION

Anshul Wadhawan

A THESIS

in

Computer and Information Science

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Computer and Information Science

2023

Supervisor of Thesis

Chris Callison-Burch

Associate Professor of Computer and Information Science

Reader of Thesis

Mark Liberman

Professor of Computer and Information Science

Graduate Group Chairperson

Swapneel Sheth, Practice Associate Professor of Computer and Information Science

SIMULTANEOUS SPEECH TO SPEECH TRANSLATION

To my beloved family,

Your unwavering support and encouragement have been the driving force behind my academic pursuits. I dedicate this thesis to you as a token of my gratitude and appreciation for all that you have done for me.

ACKNOWLEDGEMENT

I am grateful for the invaluable guidance and mentorship provided by Prof. Chris Callison-Burch throughout my thesis journey. His keen insight, patience, and constructive feedback have been instrumental in shaping my research and achieving my goals.

I also want to express my heartfelt appreciation to Liam Dugan for his tireless efforts and unwavering support throughout the project. His expertise, willingness to help, and collaborative spirit have been a great asset to me, and I could not have done this without him.

Moreover, I would like to acknowledge the role of my family and friends in providing a constant source of encouragement, motivation, and emotional support. Their belief in me and my abilities has been a driving force that kept me going during the challenging times.

In summary, I am humbled and honored to have had such a wonderful support system throughout my thesis journey, and I dedicate this achievement to all those who have played a part in making it possible.

ABSTRACT

SIMULTANEOUS SPEECH TO SPEECH TRANSLATION

Anshul Wadhawan

Chris Callison-Burch

Mark Liberman

This thesis presents our methodologies to tackle the task of simultaneous speech to speech translation - real-time conversion of speech in the source language to speech in the target language. This is a challenging task, but it has become increasingly important in today's globalized world where people from different linguistic backgrounds need to communicate with each other in real-time. To achieve this, we have developed an online speech to speech translation system with text as intermediate representations, inspired from various experiments involving offline speech to speech translation, online speech to text translation, and offline speech to spectrogram translation.

In this thesis, we also present a deeper study that formalizes the nature of hidden units and evaluates their candidacy as intermediate representations for translation tasks, as compared to text. We examine whether hidden units can provide more accurate and meaningful representations of speech that can facilitate the translation process. We conduct experiments to evaluate the effectiveness of the proposed system and compare the results with existing systems.

The online model is evaluated in terms of BLEU scores of the generated text, as well as latency metrics like average lagging. We provide a detailed analysis of the results and highlight the strengths and weaknesses of our approach. Moreover, we discuss some of the technical problems associated with the challenge, such as the handling of accents, dialects, and speaker variability, and explain how these problems can be tackled.

TABLE OF CONTENTS

ACKNOWLEDGEMENT iv

ABSTRACT v

LIST OF TABLES viii

LIST OF ILLUSTRATIONS ix

CHAPTER 1 : Introduction 1

1.1 Offline/Non-real-time Speech to Speech Translation 1

1.2 Simultaneous/Online/Real-time Speech to Speech Translation 3

CHAPTER 2 : Literature Review 5

2.1 Speech Encoders 6

2.2 Speech to Unit Translation 7

2.3 End-to-End/Direct Speech to Speech Translation 8

2.4 Attention Mechanisms 9

2.5 Simultaneous Speech to Text Translation 14

2.6 Automatic Speech Recognition 15

CHAPTER 3 : Datasets 18

3.1 Fisher Spanish-English speech translation corpus 18

3.2 Common Voice Mozilla 19

3.3 MuST-C 19

3.4 LibriSpeech 20

3.5 CoVoST and CoVoST 2 20

3.6 CVSS 21

3.7 VoxPopuli 22

CHAPTER 4 : Preliminary Experiments 23
4.1 Speech to Unit Translation + Unit to Speech Translation 23
4.2 Offline Speech to Spectrogram Translation (S2SPECT) 25
4.3 Online Speech to Text Translation (SimulST) 27
4.4 Conclusions 28

CHAPTER 5 : Simultaneous Speech to Text Translation + Text to Speech Translation . . . 30
5.1 SimulST Component: Real-time Speech-to-Text Translation with OpenAI's Whisper 31
5.2 TTS Component: Chunkwise Text-to-Speech using ElevenLabs API 32
5.3 Pipeline System Design 34
5.4 System Evaluation 35
5.5 Results 38

CHAPTER 6 : Conclusions 40
6.1 Future Work 40

BIBLIOGRAPHY 42

LIST OF TABLES

TABLE 1.1 Word Order problem in Translation 2

TABLE 4.1 SimulST results (CA = Computation Aware) 28

TABLE 5.1 Performance scores (BLEU, Average Lagging) for four policies (Offline, Greedy, Confidence-Aware (CAP), and Consensus (CP)) in SimulS2ST using Whisper Medium (769M params) (Radford et al., 2022). Optimal latency-quality trade-off shown in bold. Languages structurally similar to English exhibit better trade-off. Spanish and Russian BLEU scores approach Offline levels with minimal latency increase over Greedy policy. 37

LIST OF ILLUSTRATIONS

FIGURE 1.1 The translation process at a particular time step, as discussed in Chen et al. (2021b). 2
FIGURE 1.2 A schematic of Cascaded Speech to Speech Translation System. 2

FIGURE 2.1 Prediction of hidden cluster assignments of masked frames, in Hsu et al. (2021). 6
FIGURE 2.2 Illustration of the textless S2ST model presented in Lee et al. (2021b). 7
FIGURE 2.3 Sample mel-spectograms presented in Jia et al. (2021). 8
FIGURE 2.4 Schematics of the attention mechanisms discussed in Chiu and Raffel (2017). In this diagram, there are nodes representing the possibility of a model attending to a memory entry at a certain output time. In soft attention, the model assigns a probability to each memory entry at each output time, represented by the shade of gray on each node. The context vector is calculated as the weighted average of the memory based on these probabilities. At test time, monotonic attention examines memory entries from left-to-right, deciding whether to move on to the next memory entry or to stop and attend, shown as black nodes. The context vector is assigned to the memory entry that was attended to and at the next output time, it starts again from where it left off. MoChA uses a hard monotonic attention mechanism to determine the endpoint of the chunk to attend to, shown as nodes with bold borders. The chunk boundaries, with a window size of 3, are indicated by dotted lines. The model then performs soft attention over the chunk, with the weighting shown as the shade of gray on each node. The context vector is calculated as the weighted average of the chunk. 11
FIGURE 2.5 Schematics of the infinite lookback attention mechanism discussed in Arivazhagan et al. (2019). 12
FIGURE 2.6 Monotonic Attention (Left) versus Monotonic Multihead Attention (Right) from Ma et al. (2019). At each decoding step, MMA still has access to various encoder states. 13
FIGURE 2.7 Taken from Ma et al. (2020b), Simul-ST architecture with pre-decision module. Blue states in the figure indicate the point Simul-SST model triggers the simultaneous making process. 14
FIGURE 2.8 Taken from Radford et al. (2022), a sequence-to-sequence Transformer model is trained on many different speech processing tasks, including multilingual speech recognition, speech translation, spoken language identification, and voice activity detection. All of these tasks are jointly represented as a sequence of tokens to be predicted by the decoder, allowing for a single model to replace many different stages of a traditional speech processing pipeline. . . 17

FIGURE 4.1 Taken from Lee et al. (2021a), an illustration of the direct S2ST model with discrete units. The model consists of (1) a transformer-based speech to unit translation (S2UT) model with a speech encoder and a discrete unit decoder, (2) auxiliary tasks conditioned on the encoder, (3) a text CTC decoder conditioned on the discrete unit decoder, and (4) a vocoder separately trained to transform discrete units into waveform. 24
FIGURE 4.2 Output spectrograms of our replication of Translatotron (Jia et al., 2019) using teacher forcing. We compare the ground truth target spectrogram (left) to our model output (right) and see that the model struggles to capture the fine detail present in the output signal. 26

FIGURE 5.1 A schematic of Cascaded Simultaneous Speech to Speech Translation System. 30
FIGURE 5.2 Our cascaded SimulS2ST system follows a sequential process that involves passing speech segments from the frame buffer to OpenAI's Whisper, an offline speech to Text (ST) model. The translated text output is then generated based on the policy that has been selected for determining when to speak the output sequence. This approach ensures that the input speech frames are efficiently processed by Whisper, which produces accurate translations in real-time. The selected policy determines when the translated text is spoken, ensuring smooth and natural speech synthesis. This cascaded system allows for effective and dynamic translation of speech input, enabling practical applications in various real-world scenarios where simultaneous translation is needed. 33

CHAPTER 1

Introduction

The problem we are attempting to solve is a challenging one - simultaneous (real-time) speech to speech translation, meaning we aim to convert speech in one language (such as Spanish) to speech in another language (such as English) while the speaker is still speaking.

We can see an example of a similar system in action with a simultaneous speech to text translation system presented in Chen et al. (2021b), which is illustrated in this link[1]. Figure 1.1 shows the translation process in the demo at a particular time step. This system takes streaming speech input in one language and produces streaming text output in another language. However, our aim is to create a system that produces streaming speech output in the target language, rather than text.

Previous attempts at speech to speech translation have relied on text as an intermediate representation. This involves converting the source language speech into text, translating the text into the target language, and then converting that text back into speech using text-to-speech methods. However, this approach has limitations - it does not preserve important speech features like emotion, hesitation, interruptions, pauses, and the resulting speech output can be bland and robotic. The challenge here is to make this conversion process real-time and streaming.

The importance of this problem lies in the potential language barrier that exists between people who speak different languages. By reducing this barrier, we can enable people to communicate more easily, even if they do not know each other's language.

1.1. Offline/Non-real-time Speech to Speech Translation

Offline speech to speech translation refers to the process of translating recorded speech from one language to another without the requirement for instantaneous translation during a live conversation. In contrast to real-time speech to speech translation, which aims to provide immediate translation during a conversation, offline speech translation allows for a time lag between the original speech

[1] https://littlechencc.github.io/SimulST-demo/simulST-demo.html

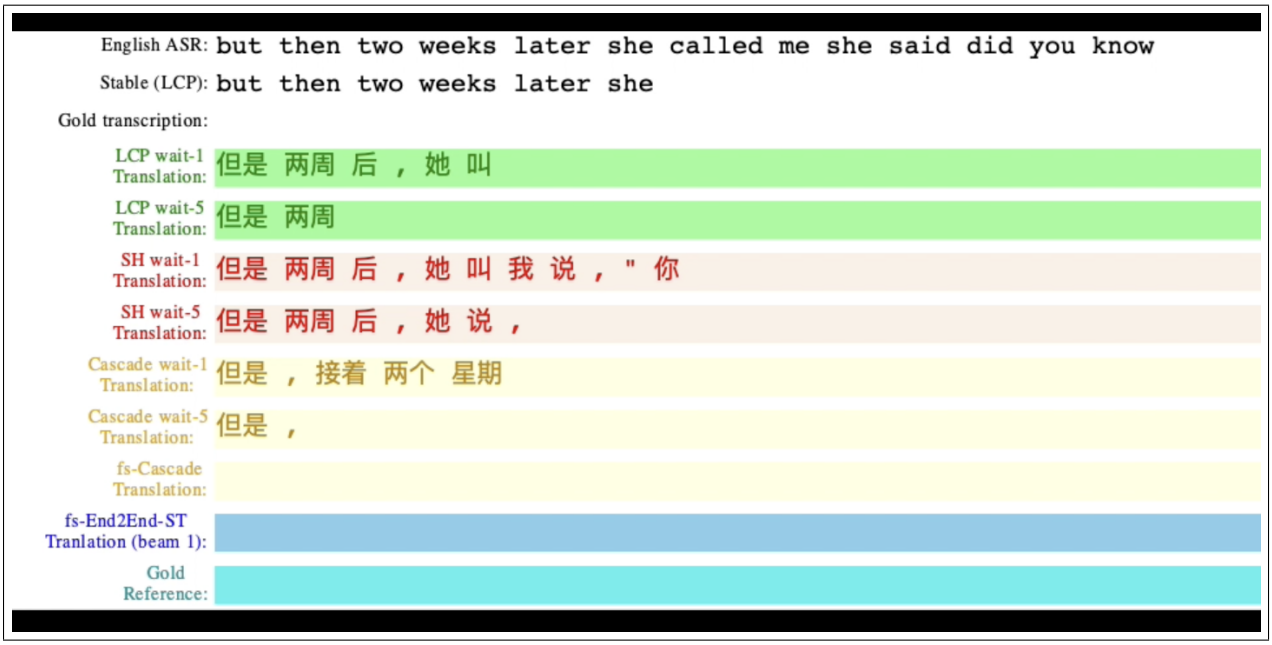

Figure 1.1: The translation process at a particular time step, as discussed in Chen et al. (2021b).

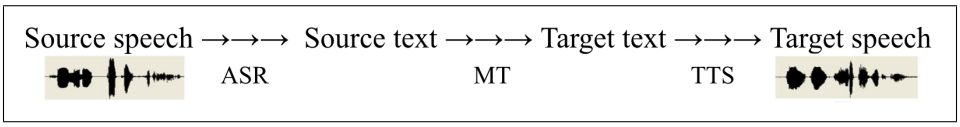

Figure 1.2: A schematic of Cascaded Speech to Speech Translation System.

and the translated output.

Offline speech to speech translation can be implemented using cascaded approaches, such as automatic speech recognition (ASR) to convert spoken words into text, machine translation (MT) to translate the text, and text-to-speech (TTS) synthesis to convert the translated text back into speech. This cascaded network is shown in Figure 1.2. However, the key difference is that the recorded speech is processed all-at-once, allowing for more time and computational resources to be devoted to the translation process, which can potentially result in higher translation accuracy. In

Original sentence: Translated sentence (in German):	"She gave him a book yesterday." "Gestern hat sie ihm ein Buch gegeben."
Original sentence: Translated sentence (in French):	"I will go to the store after work." "Après le travail, j'irai au magasin, plus tard."

Table 1.1: Word Order problem in Translation

many languages, a word occurring in the last parts of a sentence in source language, occupies the starting parts of the sentence translated in target language.

In the examples presented in Table 1.1, the phrases "yesterday" and "after work" which occur at the end of the sentence in English, are moved to the start of the sentences in German and French, respectively, resulting in a change in word order. Such sentences are better translated when they are fully available.

There are some limitations to offline speech to speech translation. One of the main drawbacks is the time lag between the original speech and the translated output, which can be a hindrance in situations where immediate translation is required for effective communication. Offline speech to speech translation may also not be suitable for interactive conversations or situations where real-time communication is crucial, such as business negotiations, customer service interactions, or emergency situations.

1.2. Simultaneous/Online/Real-time Speech to Speech Translation

Simultaneous speech to speech translation is an emerging field of research that aims to provide real-time translations of spoken language between two or more parties. Unlike offline translation, where the transcription and translation of the source language are completed before the target language output is generated, simultaneous translation involves translating the speech while it is being spoken. It involves capturing and processing the speech input on-the-fly, and providing immediate translation output to enable seamless communication between speakers of different languages.

Simultaneous speech to speech translation also presents some challenges. One of the key challenges is the need for low-latency processing to ensure that the translation is provided in near real-time, without significant delays that could disrupt the flow of conversation. Additionally, accuracy of ASR and MT technologies can impact the quality of translation, as errors or inaccuracies in speech recognition or translation can result in misunderstandings or miscommunications. Not only these, but the inability of the system to deduce the point of time where it is confident enough about the translation of spoken input and start generation, is another problem at hand. With regard to the

example sentences mentioned in Section 1.1: In cases where input sentences of this nature are short, typical solutions do not commit to speaking the output, however, if such sentences are longer, they tend to make corrections after the translation has been spoken.

Recent advancements in machine learning and natural language processing have enabled significant progress in the field of simultaneous speech to speech translation. Several approaches have been proposed, including cascaded models, end-to-end models, and hybrid models that combine both approaches. Additionally, researchers have explored the use of deep neural networks, attention mechanisms, and reinforcement learning to improve the accuracy and fluency of the translation output.

Despite the recent progress, the task of simultaneous speech to speech translation remains a challenging research problem. Further advancements are required to improve the robustness and accuracy of the translation system, making it suitable for a broader range of scenarios. The field of simultaneous speech to speech translation is rapidly evolving, and it is expected to play a significant role in breaking down language barriers and promoting cross-cultural communication.

CHAPTER 2

Literature Review

Speech to speech Translation (S2ST) has gained widespread popularity as a task in the field of natural language processing, with notable advancements in end-to-end approaches (Jia et al., 2022a) as well as textless scenarios (Lee et al., 2022). Despite these advancements, the exploration of adapting S2ST models to simultaneous applications has been relatively limited in the existing literature. This is surprising considering the evident practical applications that simultaneous S2ST can offer, such as cross-lingual voice chat and live interpretation.

While significant progress has been made in offline S2ST, where the entire input utterance is translated before generating output, simultaneous S2ST poses unique challenges due to the need for real-time translation while input is still being received. Real-world scenarios, such as cross-lingual voice chat in international settings or live interpretation in multilingual events, demand efficient and accurate simultaneous S2ST systems. However, the research in this area has been relatively under-explored compared to other aspects of machine translation, leaving ample room for further investigation and development of innovative approaches to address the specific requirements and challenges of simultaneous S2ST.

Hence, there is a pressing need to advance the state-of-the-art in simultaneous S2ST, and bridge the gap between offline S2ST and real-time applications. By leveraging the recent advancements in end-to-end and textless S2ST, and tailoring them to the unique demands of simultaneous scenarios, we can unlock the full potential of S2ST for practical and impactful use cases.

Although simultaneous speech to speech translation (S2ST) holds great potential for various applications, the development of direct end-to-end simultaneous S2ST systems remains an active research area, particularly for low-resource languages (Inaguma et al., 2022).

In this literature review, we explore several aspects of S2ST research including Speech Encoders, Speech to Unit Translation, End-to-end Speech to Speech Translation, Attention mechanisms for

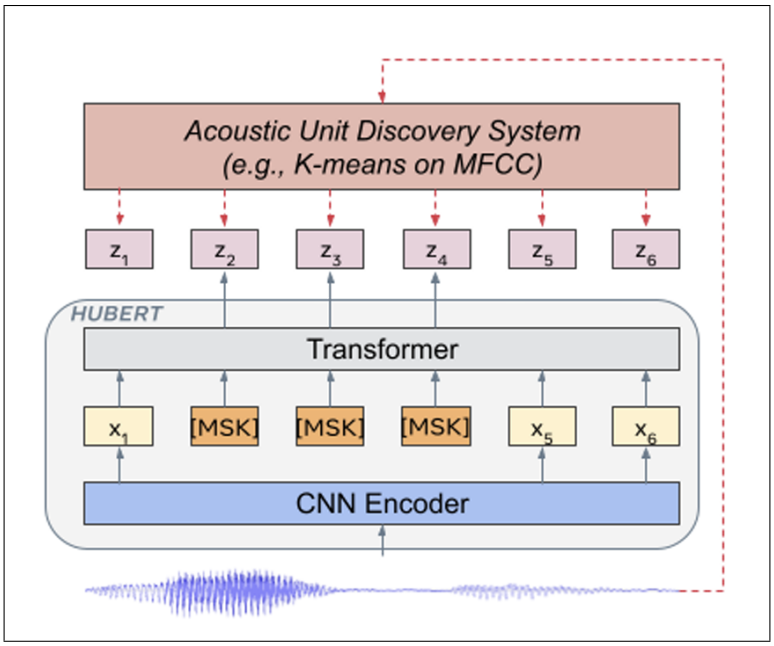

Figure 2.1: Prediction of hidden cluster assignments of masked frames, in Hsu et al. (2021).

Simultaneous Translation, Simultaneous Speech to Text Translation and Automatic Speech Recognition.

2.1. Speech Encoders

Self-supervised approaches have become increasingly popular for learning speech representations. However, there are unique challenges associated with such approaches, such as the presence of multiple sound units in each input utterance, the absence of a lexicon of input sound units during the pre-training phase, and the variable lengths of sound units without explicit segmentation.

To tackle these challenges, Hsu et al. (2021) proposed the Hidden-Unit BERT (HuBERT) approach for self-supervised speech representation learning. The method leverages an offline clustering step to provide aligned target labels for a BERT-like prediction loss (Devlin et al., 2019), which considers only the prediction of the masked tokens and ignores the prediction of the non-masked ones. By applying the prediction loss over masked regions only, the model is forced to learn a combined acoustic and language model over the continuous inputs.

A crucial aspect of the HuBERT approach is that it relies primarily on the consistency of the unsu-

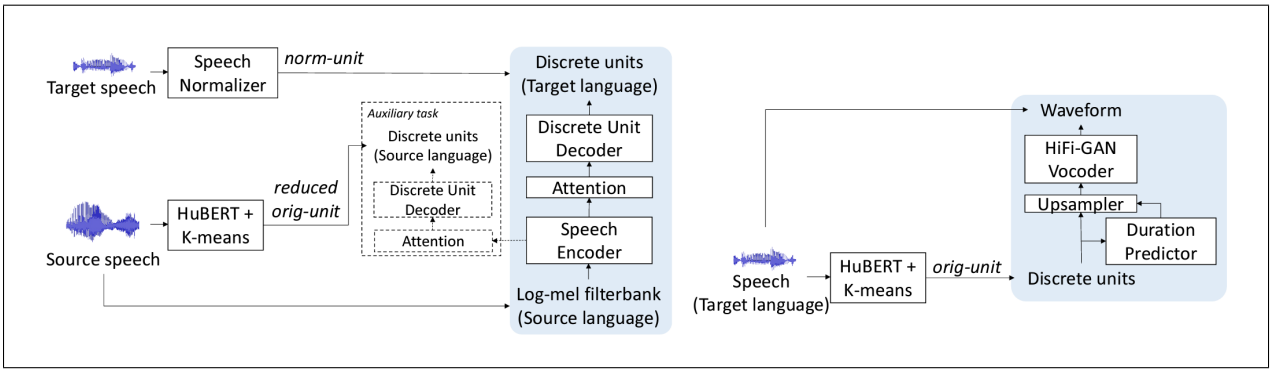

Figure 2.2: Illustration of the textless S2ST model presented in Lee et al. (2021b).

pervised clustering step rather than the intrinsic quality of the assigned cluster labels. Specifically, the HuBERT model starts with a simple k-means teacher of 100 clusters and utilizes two iterations of clustering. It either matches or surpasses wav2vec 2.0 (Baevski et al., 2020b) performance on the Librispeech (960h) and Libri-light (60,000h) benchmarks with 10min, 1h, 10h, 100h, and 960h fine-tuning subsets.

The HuBERT approach has been shown to be highly effective in reducing Word Error Rate (WER) on the dev-other and test-other evaluation subsets. In particular, using a 1B parameter model, HuBERT shows up to 19% and 13% relative WER reduction on these more challenging subsets. We use HuBERT as the speech encoder in one of our preliminary experiments, in Chapter 4.1.

2.2. Speech to Unit Translation

In recent years, there has been significant progress in the field of speech to speech translation, with many researchers exploring ways to improve the quality and efficiency of these systems. In this regard, Lee et al. (2021a) present a speech to unit translation (S2UT) model that leverages self-supervised learning techniques to overcome the limitations of existing systems. Specifically, they use HuBERT as a self-supervised discrete speech encoder to encode the target speech into discrete representations, and then train a sequence-to-sequence speech to unit translation (S2UT) model to predict these representations.

In another work on similar lines, Lee et al. (2021b) build off the work presented in Lee et al. (2021a) by introducing a self-supervised unit-based speech normalization technique. This technique fine-

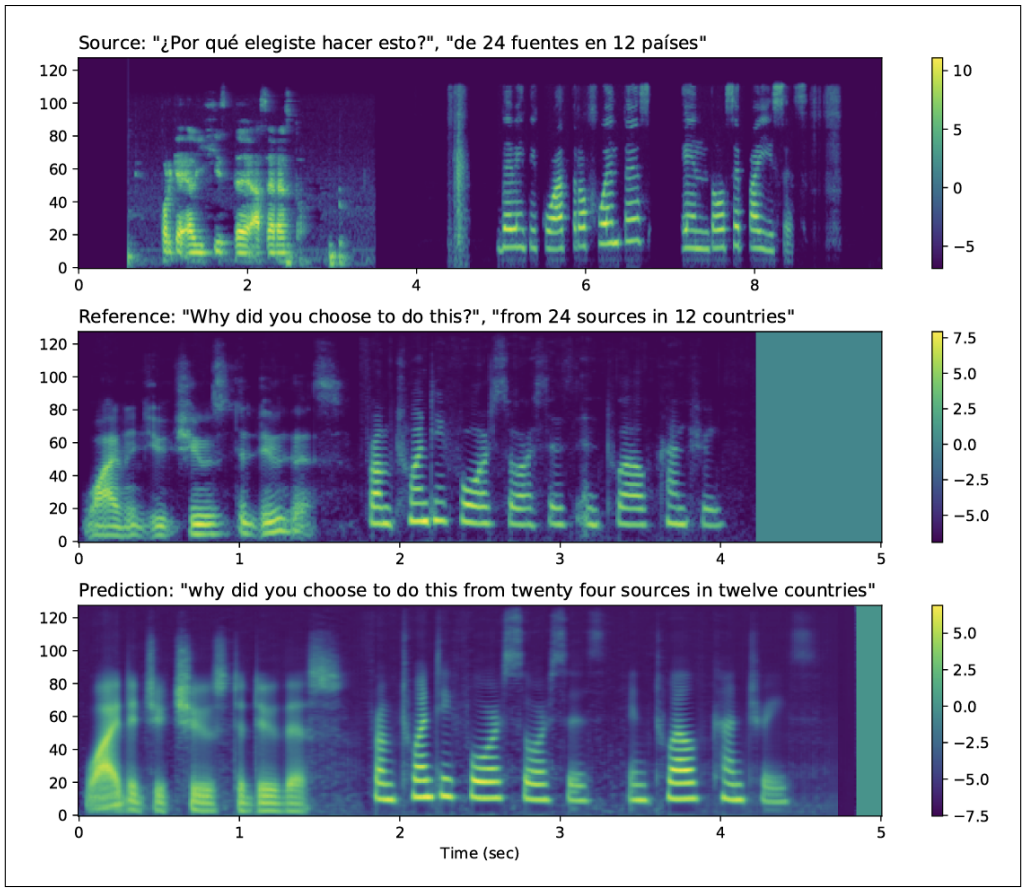

Figure 2.3: Sample mel-spectograms presented in Jia et al. (2021).

tunes a pre-trained speech encoder with paired audios from multiple speakers and a single reference speaker to reduce the variations due to accents, while preserving the lexical content. The authors train their system on real-world S2ST data, which includes multiple speakers and various accents, to better model the multi-speaker target speech.

2.3. End-to-End/Direct Speech to Speech Translation

Despite the challenging nature of fully end-to-end speech to speech translation, there have been recent advances in this area. Jia et al. (2019) and Jia et al. (2021) proposed Translatotron, an end-to-end speech to speech translation system that utilizes a sequence-to-sequence model with attention. The network is trained end-to-end, learning to map speech spectrograms from the source language into target spectrograms in another language, corresponding to the translated content in a different canonical voice. They also demonstrate the ability to synthesize translated speech using

the voice of the source speaker.

Unlike other methods, this approach does not rely on intermediate representations such as text or phonemes. Instead, it directly translates speech in one language to speech in another language, allowing for more detailed information to be captured in the input and output signals, as well as doing so with lower latency. However, training end-to-end models is more difficult due to the sparsity of training data and the complexity of modeling the speech signal.

To evaluate the performance of this approach, experiments were conducted on two Spanish-to-English speech translation datasets. Findings show that the proposed model slightly underperforms a baseline cascade of a direct speech to text translation model and a text-to-speech synthesis model. Despite the slight performance difference, the results demonstrate the feasibility of this approach on the challenging task of simultaneous speech to speech translation. We experiment with a modified version of this Translatotron architecture in Chapter 4.2.

2.4. Attention Mechanisms

Recurrent neural network (RNN) models equipped with attention mechanisms have emerged as a powerful solution for a wide spectrum of sequence-to-sequence problems in various domains, including natural language processing, speech recognition, and machine translation (Liu and Lane, 2016; Merity, 2019; Hong, 2017). These attention mechanisms allow the model to dynamically focus on different parts of the input sequence when generating each element in the output sequence, resulting in highly expressive and context-aware predictions.

However, despite their effectiveness, traditional soft attention mechanisms suffer from a drawback: they require scanning the entire input sequence at each time step during inference, which can be computationally expensive and impractical for online settings. This is because the soft attention mechanism computes a weighted sum of the input sequence representations, where the weights are obtained through a softmax operation that considers the entire input sequence. As a result, the time complexity of soft attention mechanisms grows quadratically with the length of the input sequence, which can become a bottleneck for real-time applications.

In the next few sections, we discuss various forms of attention mechanisms that were elemental in our understanding of simultaneous speech to speech translation systems.

2.4.1. Monotonic Attention

To address the above discussed computation limitation, Raffel et al. (2017) propose an innovative approach that leverages the insight that alignment between input and output sequence elements is often monotonic in many real-world problems. In other words, the attention should typically be focused on consecutive elements in the input sequence as the model generates the corresponding elements in the output sequence. Based on this insight, the method proposed by Raffel et al. (2017) propose an end-to-end differentiable method for learning monotonic alignments, which allows for online computation of attention in linear time during inference.

This method enables the model to efficiently compute attention weights based on the alignment between input and output sequence elements in a monotonic manner, without the need to scan the entire input sequence at each time step. This not only reduces the computational overhead but also makes the model more amenable for online and real-time applications where low-latency is crucial.

The proposed method has been shown to be effective in multiple sequence-to-sequence tasks, including sentence summarization, machine translation, and online speech recognition. Experimental results demonstrate that this method achieves competitive performance compared to existing sequence-to-sequence models while offering the advantage of online and linear-time computation of attention.

2.4.2. Monotonic Chunkwise Attention

An approach proposed by Chiu and Raffel (2017), Monotonic Chunkwise Attention (MoChA), introduces an adaptive strategy for addressing the above discussed limitations of soft attention mechanisms in online settings. MoChA dynamically splits the input sequence into smaller chunks, and soft attention is computed over these chunks instead of the entire input sequence. This chunkwise attention computation allows for more efficient processing during both training and inference stages.

One key advantage of MoChA is its compatibility with standard backpropagation, enabling ef-

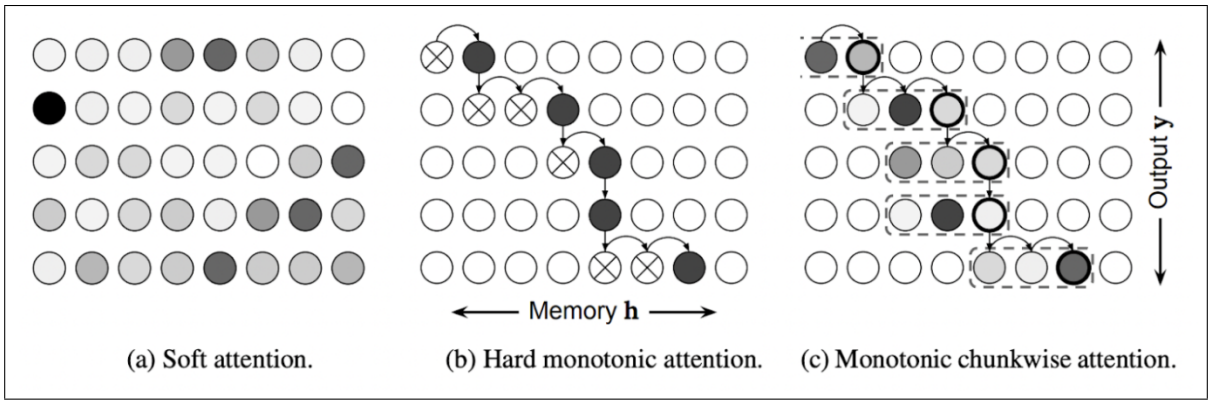

Figure 2.4: Schematics of the attention mechanisms discussed in Chiu and Raffel (2017). In this diagram, there are nodes representing the possibility of a model attending to a memory entry at a certain output time. In soft attention, the model assigns a probability to each memory entry at each output time, represented by the shade of gray on each node. The context vector is calculated as the weighted average of the memory based on these probabilities. At test time, monotonic attention examines memory entries from left-to-right, deciding whether to move on to the next memory entry or to stop and attend, shown as black nodes. The context vector is assigned to the memory entry that was attended to and at the next output time, it starts again from where it left off. MoChA uses a hard monotonic attention mechanism to determine the endpoint of the chunk to attend to, shown as nodes with bold borders. The chunk boundaries, with a window size of 3, are indicated by dotted lines. The model then performs soft attention over the chunk, with the weighting shown as the shade of gray on each node. The context vector is calculated as the weighted average of the chunk.

ficient training of the model. Furthermore, at test time, MoChA allows for online decoding in linear time, mitigating the quadratic time complexity issue associated with traditional soft attention mechanisms. To illustrate the MoChA mechanism, Figure 2.4 presents a schematic of the model architecture.

To validate the effectiveness of MoChA, experiments were conducted on online speech recognition tasks, and results demonstrate that MoChA achieves the then state-of-the-art performance, comparable to a model that uses an offline soft attention mechanism. This showcases the capability of MoChA to deliver highly accurate results while maintaining online processing efficiency.

Moreover, MoChA was evaluated on document summarization tasks, where monotonic alignments are not expected. Surprisingly, experiments revealed that MoChA outperforms a baseline monotonic attention-based model, indicating its potential for improving performance in scenarios where

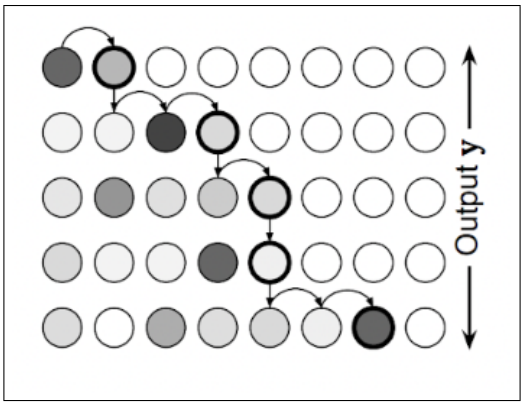

Figure 2.5: Schematics of the infinite lookback attention mechanism discussed in Arivazhagan et al. (2019).

monotonic alignments may not be present.

2.4.3. Monotonic Infinite Lookback Attention

Arivazhagan et al. (2019) propose a novel simultaneous translation system that incorporates an adaptive schedule jointly with a neural machine translation (NMT) model. Building upon the concept of Monotonic Chunkwise Attention (MoChA) (Chiu and Raffel, 2017), which adaptively splits the input sequence into small chunks for soft attention computation, this approach takes it one step further. Monotonic Infinite Lookback (MILk) attention allows the model to perform soft attention not only within the current chunk, but also over the encoder states of the entire source sentence, providing the model with an "infinite lookback" capability. Figure 2.5 presents the schematics corresponding to the monotonic infinite lookback attention mechanism.

The MILk attention mechanism in the proposed system includes both a hard, monotonic attention head for scheduling the reading of the source sentence, and a soft attention head that extends back to the beginning of the source sentence. This unique combination allows the model to dynamically adjust its attention mechanism based on the current input and the generated translations, while also considering information from earlier parts of the source sentence. The adaptive schedule learned by MILk enables the model to arrive at favorable latency-quality trade-offs, outperforming wait-k strategy for various latency values.

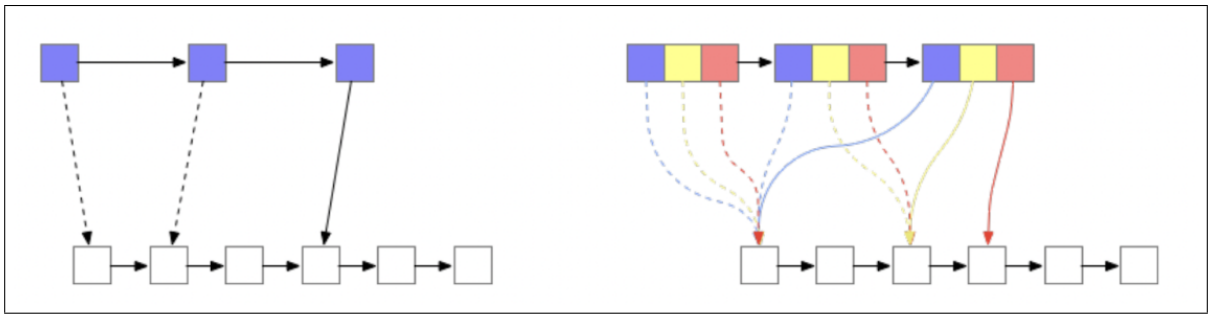

Figure 2.6: Monotonic Attention (Left) versus Monotonic Multihead Attention (Right) from Ma et al. (2019). At each decoding step, MMA still has access to various encoder states.

2.4.4. Monotonic Multihead Attention

The monotonic multihead attention mechanism proposed by Ma et al. (2019) extends the monotonic attention mechanism to multihead attention. Multihead attention is a mechanism that enables the model to focus on different parts of the input sequence by projecting the input sequence into several subspaces, which are then processed independently. Each head is able to set attention to different positions, so that it can attend to previous states while reading each new token.

In contrast, the hard monotonic model, which only employs a single attention head, lacks the ability to attend to different positions simultaneously. This can result in the loss of previous information at the attention layer as the model moves on to process new tokens. The introduction of multiple attention heads in the MMA model overcomes this limitation, as it can adjust the speed of each head on-the-fly. Some heads read new inputs, while others can stay in the past to retain the source history information.

Figure 2.6 presents the schematics corresponding to the monotonic multihead attention mechanism proposed by Ma et al. (2019). The figure shows the different attention heads attending to different parts of the input sequence, with some heads focusing on the past states while others focus on the current state. The output of each head is then concatenated and passed through a linear layer to produce the final attention output.

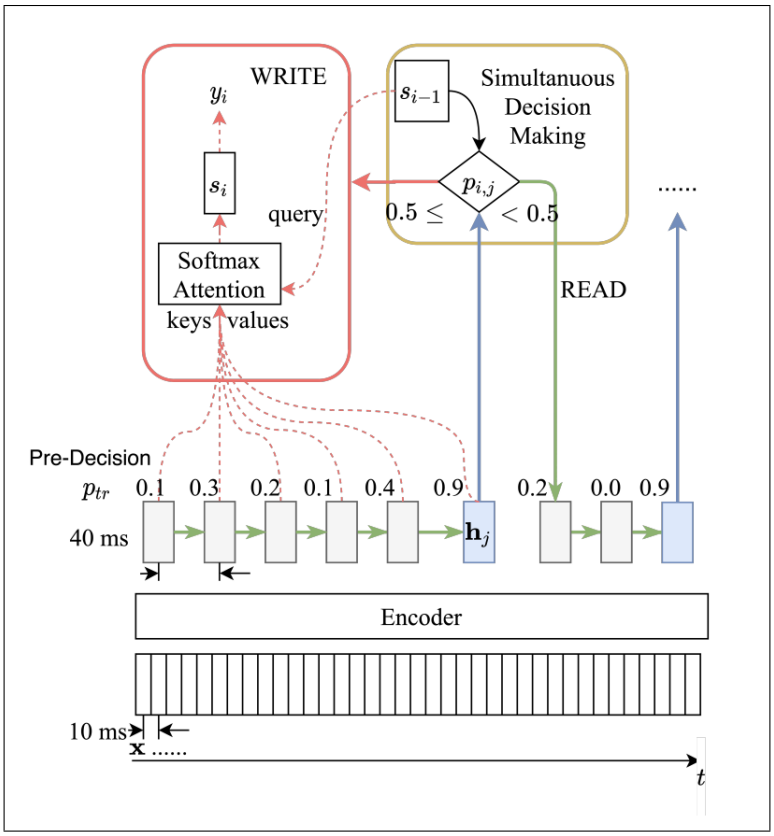

Figure 2.7: Taken from Ma et al. (2020b), Simul-ST architecture with pre-decision module. Blue states in the figure indicate the point Simul-SST model triggers the simultaneous making process.

2.5. Simultaneous Speech to Text Translation

In recent years, significant progress has been made in simultaneous text translation and end-to-end speech translation as independent tasks. However, there has been limited research on combining these tasks together. Ma et al. (2020b) investigate how to adapt simultaneous text translation methods, such as wait-k and monotonic multihead attention, to end-to-end simultaneous speech translation by introducing a pre-decision module, which guides how encoder states are grouped into meaningful units prior to making a READ/WRITE decision.

Specifically, the pre-decision module calculates the trigger probabilities p_{tr} based on the current encoder states and their corresponding features. If the calculated p_{tr} value for a given encoder state is greater than 0.5, it indicates that a simultaneous decision should be made at that point. On the other hand, if p_{tr} is less than or equal to 0.5, it suggests that the model should continue reading

new frames to gather more information before making a decision. The Simul ST architecture is presented in Figure 2.7.

A detailed analysis of the trade-offs between latency and translation quality when combining fixed and flexible pre-decision strategies with fixed and flexible policies, is provided in their work.

To evaluate the performance of the proposed approach, they introduce a novel computation-aware latency metric adapted from Average Lagging, which takes into account the computational cost of the translation process. This metric allows quantification of the efficiency of the approach in terms of computation and real-time translation performance.

2.6. Automatic Speech Recognition

Automatic Speech Recognition (ASR) is the field of research and technology that focuses on developing systems capable of automatically converting spoken language into written text. ASR has a wide range of applications, including transcription services, voice assistants, speech to text translation, and more. ASR research involves designing and training algorithms to accurately recognize and transcribe speech, overcoming challenges such as background noise, accents, dialects, speaking rate variations, and language-specific characteristics.

Whisper, introduced by Radford et al. (2022), is a cutting-edge automatic speech recognition (ASR) system that has been trained on an extensive dataset of 680,000 hours of multilingual and multitask supervised data collected from the web. As highlighted in their work, this massive and diverse dataset has resulted in improved robustness to various challenges, such as accents, background noise, and technical language. Additionally, Whisper is capable of transcribing speech in multiple languages and even translating them into English. The model and inference code of Whisper has been open sourced, providing a solid foundation for the development of practical applications and further research in the field of robust speech processing.

The architecture of Whisper is designed as a straightforward end-to-end approach, implemented as an encoder-decoder Transformer. The input audio is divided into 30-second chunks, converted into a log-Mel spectrogram, and then fed into an encoder. A decoder is trained to generate the

corresponding text caption, which is interspersed with special tokens that guide the single model to perform various tasks, such as language identification, phrase-level timestamps, multilingual speech transcription, and speech translation to English.

Compared to other existing approaches that often rely on smaller and more closely paired audio-text training datasets or unsupervised audio pretraining, Whisper stands out as it was trained on a large and diverse dataset collected from the web, (Chan et al., 2021; Galvez et al., 2021; Chen et al., 2021a). Unlike models that are fine-tuned specifically for benchmark datasets such as LibriSpeech (Panayotov et al., 2015b), which is known for its competitiveness in speech recognition, Whisper may not outperform them when training and testing data are drawn from the same distribution. However, when evaluating Whisper's zero-shot performance across diverse datasets, it exhibits significantly improved robustness, making 50% fewer errors compared to those specialized models, (Baevski et al., 2020a, 2021; Zhang et al., 2022).

Approximately one third of the audio dataset used to train Whisper comprises non-English data, and the model is trained to transcribe the audio in the original language or translate it to English alternatively. It was observed that this approach is highly effective in learning speech to text translation, surpassing the supervised state-of-the-art (SOTA) performance on CoVoST2 (Wang et al., 2020a) to English translation in a zero-shot setting. We use Whisper as the ASR model in our final experiment described in Chapter 5.

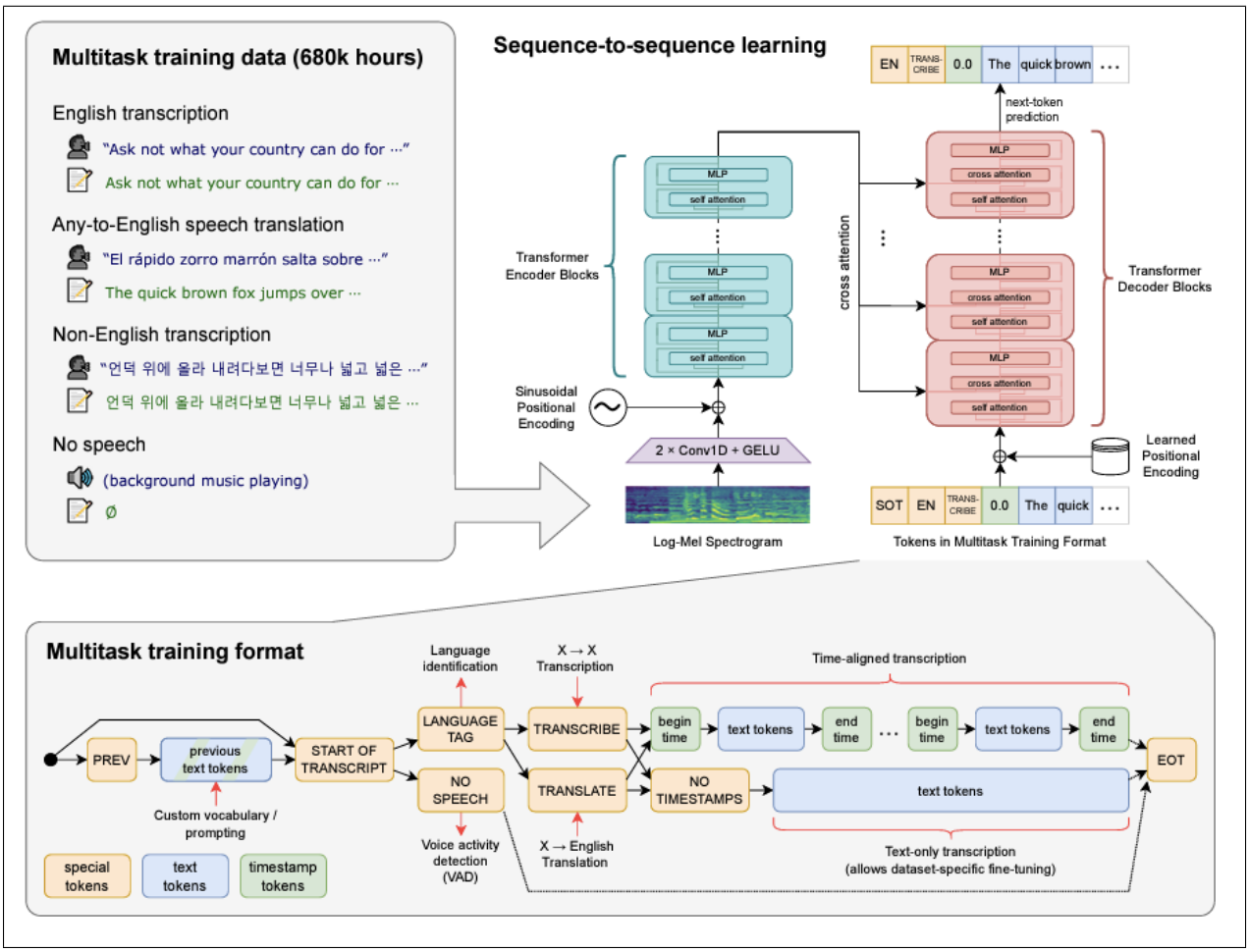

Figure 2.8: Taken from Radford et al. (2022), a sequence-to-sequence Transformer model is trained on many different speech processing tasks, including multilingual speech recognition, speech translation, spoken language identification, and voice activity detection. All of these tasks are jointly represented as a sequence of tokens to be predicted by the decoder, allowing for a single model to replace many different stages of a traditional speech processing pipeline.

CHAPTER 3

Datasets

This chapter provides an overview of multiple datasets that are relevant for the task of speech to speech translation. The datasets have been used for training and evaluating various models, and have contributed significantly to the advancement of speech translation research. We provide a brief description of each dataset along with its properties and size.

3.1. Fisher Spanish-English speech translation corpus

The Fisher and CALLHOME Spanish-English Speech Translation dataset (Post et al., 2013), developed at Johns Hopkins University, provides a valuable resource for research in automatic speech translation. This dataset contains English reference translations and speech recognizer output, complementing the LDC Fisher Spanish and CALLHOME Spanish audio and transcript releases. The dataset comprises approximately 38 hours of speech, with defined training, development, and held-out test sets, making it a rich source for training and evaluating ASR and translation models.

The source data for this dataset are the Fisher Spanish and CALLHOME Spanish corpora, which were developed by LDC. The Fisher Spanish dataset consists of 819 transcribed conversations on various topics primarily between strangers, resulting in approximately 160 hours of speech aligned at the utterance level, with 1.5 million tokens. The CALLHOME Spanish corpus comprises 120 transcripts of spontaneous conversations primarily between friends and family members, resulting in approximately 20 hours of speech aligned at the utterance level, with just over 200,000 words (tokens) of transcribed text.

Translations for this dataset were obtained through crowdsourcing using Amazon's Mechanical Turk, after which the data was split into training, development, and test sets. The CALLHOME data set already defines its own data splits, organized into train, devtest, and evltest, which were retained in this dataset. For the Fisher material, four additional data splits were produced, including a large training section and three test sets. These test sets correspond to different portions of the

data where four translations exist, providing a diverse range of translation variations for evaluation purposes.

3.2. Common Voice Mozilla

Common Voice (Ardila et al., 2020) is an open-source, multi-language dataset developed by Mozilla to train and evaluate automatic speech recognition (ASR) and other related models. It is one of the largest and most diverse speech datasets available publicly, with over 9,283 recorded hours and 7,335 validated hours in 60 languages, as of the latest release. The dataset consists of unique MP3 files and their corresponding text transcriptions.

One of the unique features of the Common Voice dataset is that it includes demographic metadata such as age, sex, and accent, which can be used to analyze the performance of models based on different demographic factors. This information can be particularly useful for developing more inclusive and diverse voice technologies that work well for everyone regardless of their background.

3.3. MuST-C

Containing 385 hours from Ted talks for speech translation, the MuST-C Dataset (Di Gangi et al., 2019a) is a multilingual speech translation corpus. It was created to facilitate the training of end-to-end systems for SLT from English into several languages, including Dutch, French, German, Italian, Portuguese, Romanian, Russian, and Spanish. At least 385 hours of audio recordings from English TED Talks are included in MuST-C for each target language. These recordings are automatically aligned at the sentence level with their manual transcriptions and translations, ensuring the corpus's quality and size.

To ensure high quality and large size, MuST-C was created with a focus on speaker variety and coverage of various topics and languages. The corpus was constructed based on English TED Talks, which provide an excellent source of material for creating a corpus for SLT due to their manually-transcribed and translated content. The talks feature a diverse range of speakers discussing an array of topics, including those related to business, science, and entertainment. These features make MuST-C a valuable resource for training end-to-end systems for speech translation in multiple

languages.

3.4. LibriSpeech

The LibriSpeech corpus (Panayotov et al., 2015a), which is part of the LibriVox[2] project, comprises around 1,000 hours of audiobooks, with the majority of the collection consisting of audiobooks from Project Gutenberg[3]. The corpus was designed to provide a collection of English read speech that is suitable for training speech recognition systems. To achieve this, LibriSpeech automatically aligns and segments the audiobook read speech with the corresponding book text. This process includes filtering out segments with noisy transcripts to ensure the quality of the resulting corpus.

3.5. CoVoST and CoVoST 2

Most existing datasets for speech to text translation involve language pairs with English as the source language, specific domains, or low resource languages. Wang et al. (2020a) introduce CoVoST, a multilingual speech to text translation corpus that includes 11 languages translated into English. CoVoST is unique in its diversity, comprising over 11,000 speakers and over 60 accents, making it a valuable resource for multilingual ASR research. They describe the methodology used to create the dataset and provide empirical evidence of the data quality. Additionally, they present initial benchmarks, including what we believe to be the first end-to-end many-to-one multilingual models for spoken language translation.

CoVoST V2 (Wang et al., 2020b) represents a significant expansion of the CoVoST dataset, which is designed for multilingual speech to text translation (ST). With this new release, CoVoST V2 offers a significantly large multilingual ST dataset, allowing for translation from 21 languages into English, as well as from English into 15 languages.

CoVoST V1 already boasted an impressive collection of speech data, featuring over 11,000 speakers and 60 accents across languages such as French, German, Dutch, Russian, Spanish, Italian, Turkish, Persian, Swedish, Mongolian, and Chinese, totaling to 708 hours of speech. In CoVoST V2, the dataset was expanded to include additional languages such as Welsh, Catalan, Slovenian, Estonian,

[2]https://librivox.org/
[3]https://www.gutenberg.org/

Indonesian, Arabic, Tamil, Portuguese, Latvian, and Japanese, resulting in a substantial increase in the total amount of speech data available. CoVoST V2 encompasses a staggering 2,900 hours of speech, making it one of the most comprehensive multilingual speech to text translation datasets available for research and development purposes.

3.6. CVSS

CVSS (Jia et al., 2022b) is an extensive and diverse multilingual-to-English speech to speech translation corpus that combines the strengths of the Common Voice speech corpus and the CoVoST 2 speech to text translation corpus. CVSS encompasses sentence-level parallel speech to speech translation pairs from 21 different languages into English, making it a valuable resource for multilingual speech to speech translation research. The translation speech in CVSS is synthesized using advanced text-to-speech (TTS) models trained on the LibriTTS (Zen et al., 2019) corpus, and it comes in two distinct versions: CVSS-C and CVSS-T.

CVSS-C features translation speeches in a single canonical speaker's voice, despite being synthetic, these speeches are highly natural and clean, with consistent speaking style. This makes it suitable for modeling the target speech and generating high-quality translation speech that can be used in user-facing applications. On the other hand, CVSS-T includes translation speeches in voices transferred from the corresponding source speeches, ensuring similar voices on both sides of the translation pairs, even though they are in different languages. This makes CVSS-T ideal for building models that preserve the speaker's voices during translation.

CVSS also includes the source speeches from the Common Voice corpus, making it a comprehensive multilingual speech to speech translation dataset with approximately 1,900 hours of speech for each version. In addition to translation speech, CVSS provides normalized translation text that matches the pronunciation in the translation speech, including numbers, currencies, acronyms, and more. This text can be utilized for both model training and standardized evaluation purposes.

3.7. VoxPopuli

VoxPopuli (Wang et al., 2021) is a speech dataset that consists of unlabelled speech data, transcribed speech data, and speech to speech interpretation data. The dataset includes 400K hours of unlabelled speech data for 23 languages, 1.8K hours of transcribed speech data for 16 languages, 17.3K hours of speech to speech interpretation data for 15x15 directions, and 29 hours of transcribed speech data of non-native English intended for research in ASR for accented speech (15 L2 accents).

The raw data used in VoxPopuli is collected from the 2009-2020 European Parliament event recordings. This dataset has a diverse range of languages. The unlabelled speech data can be used for pre-training large neural networks for speech recognition or speech synthesis tasks. The transcribed speech data can be used for training speech recognition models, and the speech to speech interpretation data can be used for training multilingual speech to speech translation models.

The 29 hours of transcribed speech data of non-native English is especially useful for research in accented speech recognition, which is a challenging problem due to the variations in pronunciation, intonation, and speech rate. This dataset is unique in that it provides speech data from 15 L2 accents, which can help researchers develop more robust ASR models that can handle various accents.

CHAPTER 4

Preliminary Experiments

This chapter provides an overview of three small-scale toy experiments. Chapter 4.1 talks about an offline speech to speech translation system which utilises hidden units as intermediate representations (Hsu et al., 2021; Kong et al., 2020), aimed at evaluating the nature of hidden units and their capability to preserve input speech features like prosidy, inflection, pauses and emotion. Chapter 4.2 describes an offline end-to-end speech to speech translation system (Jia et al., 2019, 2021), aimed at comparing their accuracy and latency with those from cascaded architectures. Chapter 4.3 puts forward an online speech to text translation system (Ma et al., 2020b), aimed at understanding the various building blocks of simultaneous translation systems like wait-k policies, pre-decision modules and monotonic attention(Raffel et al., 2017).

4.1. Speech to Unit Translation + Unit to Speech Translation

4.1.1. Experiments

In this experiment, we aim to evaluate the effectiveness of the offline speech to speech translation system by analyzing the speaker characteristics of the translated speech. We start by manually selecting a set of audio samples in Spanish (the source language) and pass them through the offline speech to speech translation system to convert them into English (the target language). To perform this translation, the speech segments in the source language are first converted into discrete units using the HuBERT (Hsu et al., 2021) model, which serves as a self-supervised discrete speech encoder. The resulting discrete units are then fed into a vocoder network, in this case a HiFIGAN (Kong et al., 2020) network, which generates the translated speech in the target language.

Rather than doing a quantitative evaluation, we did a qualitative analysis. To evaluate the quality of the translation and the preservation of speech features, we record speech segments in Spanish with varying prosody, pauses, and voices. Specifically, we collect speech samples from speakers with different accents and speaking styles to ensure that the system can generalize well to diverse speakers. We also record speech with different emotions and moods, such as happy, sad, and angry,

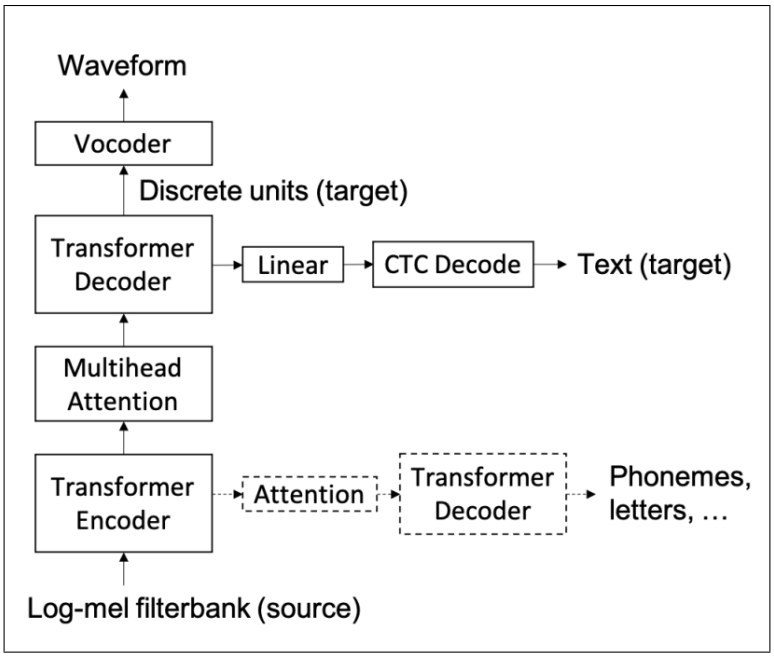

Figure 4.1: Taken from Lee et al. (2021a), an illustration of the direct S2ST model with discrete units. The model consists of (1) a transformer-based speech to unit translation (S2UT) model with a speech encoder and a discrete unit decoder, (2) auxiliary tasks conditioned on the encoder, (3) a text CTC decoder conditioned on the discrete unit decoder, and (4) a vocoder separately trained to transform discrete units into waveform.

to test the robustness of the system.

In the first round of experiments, we use speech segments from the Common Voice (Ardila et al., 2020) dataset as the source language data. We perform a subjective evaluation by asking human evaluators to rate the quality of the translated speech. Finally, we analyze the speaker characteristics of the translated speech to determine whether the system preserves important aspects of the speaker's voice, such as their tone, pitch, and intonation.

Figure 4.1 illustrates the speech to speech translation (S2ST) model with discrete units used in our experiments. This model is based on the self-supervised discrete speech encoder HuBERT (Hsu et al., 2021), which encodes the speech signal into a sequence of discrete units. The discrete units are then fed into a sequence-to-sequence speech to unit translation (S2UT) model, which generates the corresponding sequence of discrete units in the target language. Finally, the HiFIGAN (Kong et al., 2020) vocoder network converts the discrete units into speech in the target language.

4.1.2. Results

For the above experiment, we evaluate the performance of our speech to unit translation system on speech samples from the Common Voice dataset and manually recorded audios. The results are presented in this spreadsheet[4]. Our study led us to make the following observations:

- The system trims silence in the input, and it is no longer present in the output.
- Background noise in the input does not worsen the output.
- Bad audio quality in the input worsens the output.
- The output does not change significantly when there is inflection in the input.
- The output does not change significantly when the input introduces emotion.
- The output remains the same even when the input is spoken by a novice.

The system was found to effectively trim silence from the input and ignore background noise. However, the quality of the input audio significantly affected the output. The system was also observed to be immune to variations in inflection and emotion in the input and did not vary significantly based on the experience of the speaker. The study sheds light on the strengths and limitations of the speech-to-unit translation system and hidden units in general, providing insights that can guide future research and development in this area.

4.2. Offline Speech to Spectrogram Translation (S2SPECT)

4.2.1. Experiments

The offline speech to spectrogram translation system (S2SPECT) is a promising experimental approach that aims to improve the performance of offline and online S2UT systems by better capturing speaker identity and prosody with lower latency. In this study, we modified the Translatotron architecture proposed by Jia et al. (2019) by replacing the original BiLSTM with Transformer-based encoders and decoders (Vaswani et al., 2017). This modification was made with the expectation

[4]https://docs.google.com/spreadsheets/d/1BS9hJA-OhGlaYoFAvFNFbtPOO7EfjdlIl7VX2Cdggbw

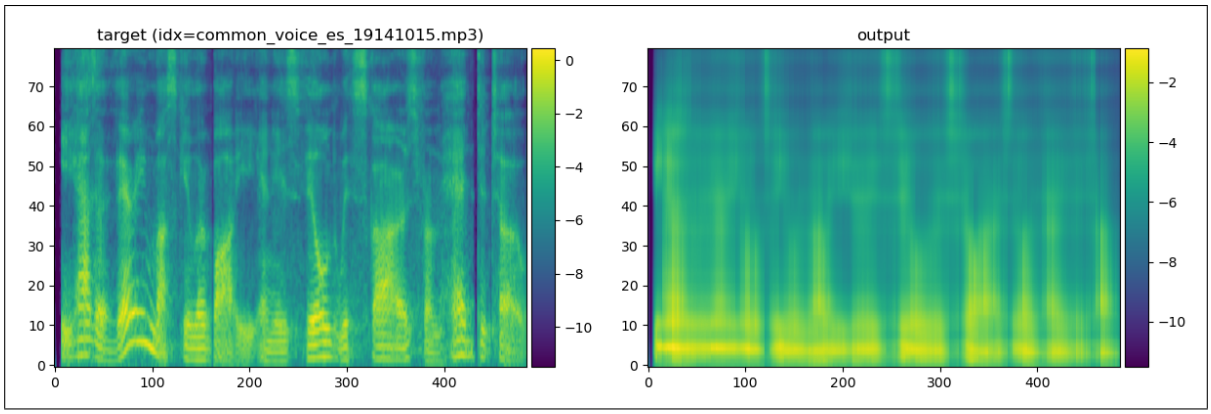

Figure 4.2: Output spectrograms of our replication of Translatotron (Jia et al., 2019) using teacher forcing. We compare the ground truth target spectrogram (left) to our model output (right) and see that the model struggles to capture the fine detail present in the output signal.

that it would lead to improvements in speaker identity and prosody modeling, as Transformer-based models have been shown to perform better than BiLSTMs in a range of natural language processing tasks (Devlin et al., 2019; Yang et al., 2020; Liu et al., 2019).

To train the S2SPECT model, we used the Spanish to English task and the Common Voice Synthetic Speech Dataset (CVSS) (Jia et al., 2022b). The CVSS dataset consists of 200 hours of Spanish speech and corresponding English translations, and was used to train a neural vocoder. The neural vocoder used in this study was a HiFi-GAN model (Kong et al., 2020), which is a state-of-the-art generative model that can reconstruct high-fidelity speech from a spectrogram.

4.2.2. Results

Rather than doing a quantitative evaluation, we did a qualitative analysis. The preliminary results of our offline speech to spectrogram translation system with teacher forcing (Williams and Zipser, 1989) are shown in Figure 4.2. This evaluation was done manually with multiple samples and Figure 4.2 is indicative of the overall findings. We see here that the end-to-end system is able to match the general shape of the output waveform but struggles to capture the finer detail. This could be attributed to the limited modeling capacity of the architecture used or the sparsity of the training data. To address this issue, we plan to use a more expressive model and augment our dataset.

To evaluate the quality of the reconstructed speech, we train HiFi-GAN (Kong et al., 2020) on the

teacher forced output. In our preliminary evaluation, we find that the reconstructed speech is high quality and intelligible. HiFi-GAN outputs on these spectrograms are of comparable quality to real speech. We find that the generative HiFi-GAN model is able to backfill the missing detail of the model's output spectrogram.

One important caveat of these results is that it is based on the use of teacher forcing (Williams and Zipser, 1989). When not using teacher forcing, we find that the model struggles with degeneration. Specifically, it starts off intelligible and then devolves into random noise. HiFi-GAN outputs on these spectrograms are high quality but generally unintelligible. Since the non-teacher forced setting is the one that would be used in any practical application of this technology, this is a serious shortcoming.

4.3. Online Speech to Text Translation (SimulST)

4.3.1. Experiments

To enhance our comprehension of simultaneous translation systems and tackle the engineering issues related to these systems, we carried out this experiment, which involved running an online speech to text translation system using two sample audios. We attempted to reproduce the findings of Ma et al. (2020b). The S-Transformer architecture, proposed by Di Gangi et al. (2019b), is utilized in this approach. This architecture has shown competitive performance on the MuST-C dataset. The encoder incorporates a two-dimensional attention mechanism following the CNN layers, and also includes a distance penalty to bias the attention towards short-range dependencies. The architecture has been described in detail, in Section 2.5.

Our aim was to evaluate the simultaneous behavior of such systems by examining the decoded tokens as they were being generated, in real time. To help visualize the simultaneous decoding process, we have included a recording of this experiment on this link[5].

4.3.2. Results

We evaluated the online speech to text translation system by giving in two sample audios and compared the simultaneous nature of decoded tokens as they were being generated. The performance

[5]https://drive.google.com/file/d/1Kd0pi_tkVH3uoXoNHM7sY38-CoUZfTcR

Metric	MuST-C	Manual Samples
BLEU	13.94	11.28
Average Lagging	1751.80	1631.66
Average Lagging (CA)	2338.59	48726.33
Average Proportion	0.79	0.989
Average Proportion (CA)	0.94	42.856
Differentiable Average Lagging	1987.78	1674.674
Differentiable Average Lagging (CA)	2425.27	77021.50

Table 4.1: SimulST results (CA = Computation Aware)

of the system on the MuST-C test dataset and the 2 manual samples in terms of ASR BLEU score and latency, measured in terms of Average Lagging, Average Proportion and Differentiable Average Lagging (Ma et al., 2020a), is summarized in Table 4.1. We found that the generated outputs were intelligible and sounded clear. However, we observed high latency values for computation-aware metrics due to the fact that we ran these experiments on a CPU instead of a GPU.

It is worth noting that the scores for the MuST-C test dataset presented in Table 4.1 were taken from Ma et al. (2020b). Specifically, reducing the latency of the system will be critical to making it more useful for real-time applications. Furthermore, improving the accuracy of the system's predictions would make it more reliable and effective for a broader range of use cases.

4.4. Conclusions

The first preliminary experiment that used hidden units as intermediate representations for the task of speech to speech translation, described in Section 4.1, suggested that hidden units were not sufficient to capture input speech characteristics like pauses, emotion, inflection, pitch, accent, etc. We concluded that hidden units were not suitable candidates to replace text as intermediate representation in cascaded approaches aimed at solving the task of simultaneous speech to speech translation.

In Section 4.2, the end-to-end speech to speech translation model struggling with degeneration when not using teacher forcing, and the inability to produce unintelligible outputs made us realize that the end-to-end setting is inappropriate for the use case of real time translation applications.

The above two shortcomings along with the promising results of the online speech to text translation system in Section 4.3, drew our attention to using text as an intermediate representation for the task of speech to speech translation, and we shifted our focus on making this cascaded architecture simultaneous/real-time.

CHAPTER 5

Simultaneous Speech to Text Translation + Text to Speech Translation

In this chapter, we discuss a system developed for simultaneous speech to speech translation (S2ST) with a focus on real-world applications like cross-lingual voice chat and live interpretation. Our approach circumvents the limitations of direct end-to-end methods by employing a two-step process that involves a SimulST component for converting source speech to target text, followed by a text-to-speech (TTS) component for generating target speech from the translated text. Figure 5.1 describes this cascaded architecture.

This cascaded approach offers several advantages. First, it allows us to leverage existing advancements in SimulST and TTS separately, making it more feasible to adapt and optimize each component for its specific task. This is particularly beneficial for low-resource languages where training data for direct end-to-end S2ST may be limited. Second, the cascaded architecture allows for flexibility in the choice of SimulST and TTS models, enabling us to select the best-performing models for each component independently, which may lead to improved overall translation quality. Finally, this cascaded system also provides modularity, making it easier to update or replace individual components without affecting the entire pipeline, which enhances the system's adaptability and maintainability.

Our system boasts an extensive language support, enabling translation from 57 different languages to English. One of the unique features of our system is the ability to dynamically adjust the latency of the output through tunable parameters, including four policies that determine when to generate an output sequence.

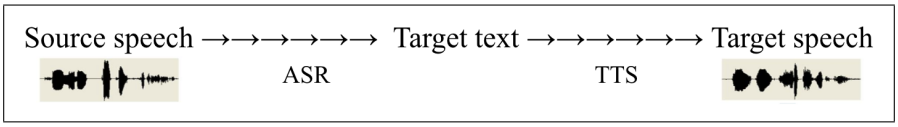

Figure 5.1: A schematic of Cascaded Simultaneous Speech to Speech Translation System.

By developing a robust cascaded SimulS2ST system, we aim to overcome the current research challenges associated with direct end-to-end simultaneous S2ST, especially for low-resource languages. Our proposed approach has the potential to significantly advance the field of simultaneous S2ST and pave the way for practical and effective real-time translation systems in a wide range of applications.

In our evaluation, we carefully analyze the trade-off between latency and translation quality using four prototyped policies for determining when to speak a given output utterance. We aim to strike the optimal balance between minimizing latency for real-time applications and maintaining high translation quality. By leveraging the capabilities of an off-the-shelf offline ST model in an online fashion, we aim to contribute to the advancement of SimulST research and development, and provide insights into the performance of different policies for determining the timing of output speech in real-time translation systems.

Our experimental results demonstrate that the policies effectively achieve accuracy levels comparable to offline S2ST, with only minimal increases in latency when compared to a Greedy (wait-k) baseline approach. We make our multi-threaded pipeline code and evaluation results publicly available to support and facilitate future research and development in Simultaneous speech to speech Translation (SimulS2ST)[6].

Section 5.1 describes the SimulST component, that is responsible for converting speech in source language to text in target language using Whisper. Section 5.2 describes the Text to Speech component that uses Eleven Labs API to convert the generated text to speech. Section 5.3 describes the System Design of the pipeline that runs the two components in unison. Section 5.4 talks about the System Evaluation of the pipeline and Section 5.5 puts forward the results.

5.1. SimulST Component: Real-time Speech-to-Text Translation with OpenAI's Whisper

Drawing inspiration from the work of Papi et al. (2022), we adopt a novel approach in our research by utilizing an existing offline speech translation (ST) model, specifically OpenAI's Whisper (Radford et al., 2022), in an online fashion to achieve accurate translations with low latency. Instead

[6]http://github.com/liamdugan/speechtospeech

of training a separate model specifically for Simultaneous speech to speech Translation (SimulST), we leverage the capabilities of an off-the-shelf ST model to handle real-time translation tasks.

Our approach involves querying the Whisper model on-the-fly during the translation process, allowing us to generate translations in near real-time as input utterances are being received. This eliminates the need for pre-processing or buffering of entire utterances before translation, which can significantly reduce the latency of the system. By utilizing an existing ST model that has been trained on a large amount of data, we benefit from the robustness and accuracy of the model without incurring the additional overhead of training a new model specifically for SimulST.

To provide a visual demonstration of the capabilities of OpenAI's Whisper, we have included a recording of our toy experiment on this link[7]. In this particular toy experiment, we focused on translating speech in Japanese to text in English.

5.2. TTS Component: Chunkwise Text-to-Speech using ElevenLabs API

The component that is responsible converting text to speech (TTS) involves several sub components that work together to produce the desired output. In this TTS component, we divide the process into three distinct parts that work in tandem to create a seamless and efficient system.

First, the writer thread, which is responsible for simulating the streaming output of the speech to text translation module. To accomplish this, we create a dummy thread that writes a string to a file, generating 5 words every second. This approach allows us to simulate the SimulST component by generating a continuous stream of text that can be converted to speech in real time. This is done to ensure modularity, and avoid errors that may propagate from the SimulST component to this component.

Second, the conversion thread, which monitors the file generated by the writer thread for new content being written. As new content is added to the file, the conversion thread converts this content to speech using the ElevenLabs API (ElevenLabs, 2023). Once the speech has been generated, it is written to a `wav` file, with a new file being created for each segment of text processed. Once the

[7] https://drive.google.com/file/d/1vXLvc1yMPWegR60eCos6CdAtp_LL3bND/view?usp=share_link

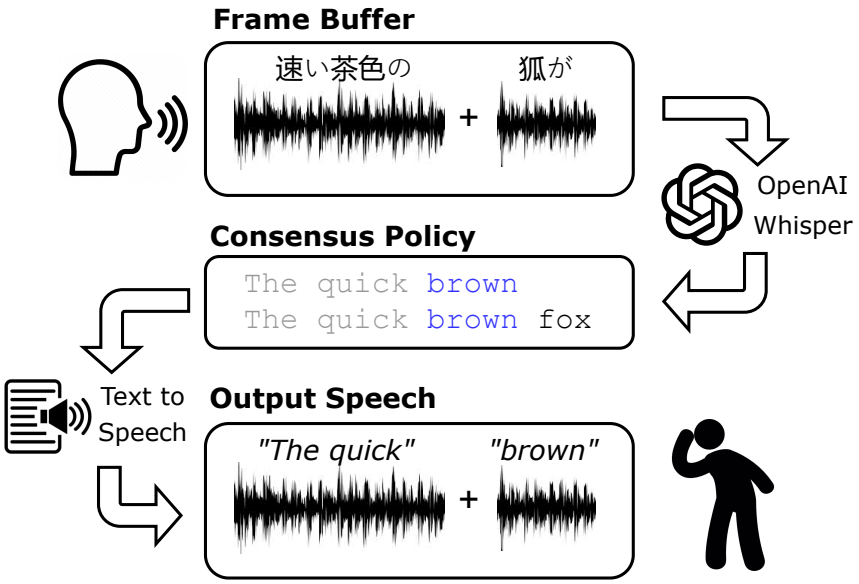

Figure 5.2: Our cascaded SimulS2ST system follows a sequential process that involves passing speech segments from the frame buffer to OpenAI's Whisper, an offline speech to Text (ST) model. The translated text output is then generated based on the policy that has been selected for determining when to speak the output sequence. This approach ensures that the input speech frames are efficiently processed by Whisper, which produces accurate translations in real-time. The selected policy determines when the translated text is spoken, ensuring smooth and natural speech synthesis. This cascaded system allows for effective and dynamic translation of speech input, enabling practical applications in various real-world scenarios where simultaneous translation is needed.

`wav` file is created, the conversion thread appends the file path to a queue for further processing.

Finally, the player thread, which monitors the above queue in a continuous loop. If the queue has any audio file paths, the player thread takes out one file path at a time and plays it through the speaker. This ensures that the audio output is generated and played back in the correct order, with no delays or interruptions.

Together, these three components work in parallel to simulate a TTS module that is efficient and effective. To help visualize the process, we have included a recording of this toy experiment. The recording is available on this link[8], and it provides a detailed demonstration of how the TTS component works and how it can be used to generate high-quality speech output from text.

5.3. Pipeline System Design

In Figure 5.2, we present a diagram illustrating the functioning of our system - a cascaded pipeline that has the SimulST component followed by the TTS component. At a high level, the system operates as follows: Let $S = s_1 \dots s_N$ denote the input sequence comprising N speech frames. In each iteration, we retrieve a chunk of size w (where $w < N$) from the input, denoted as $s_c \dots s_{c+w}$, and append it to the frame buffer $F = s_f \dots s_{f'}, s_{f'+1} \dots s_{f'+w}$. Subsequently, we provide the frame buffer F and the current spoken transcription $T = t_1 \dots t_t$ as input to the Speech to Text (ST) model, which generates the output text sequence $\hat{T} = t_{t+1} \dots t_p$. This output text sequence $\hat{T}$ is then passed to the policy $\mathcal{P}$, which determines whether or not the system should speak the generated sequence.

If the policy $\mathcal{P}$ determines that the generated text sequence $\hat{T}$ should be spoken, we utilize the Text-to-Speech (TTS) model to convert $\hat{T}$ into speech, and append $\hat{T}$ to the current transcription T, after which we clear the frame buffer. On the other hand, if the policy $\mathcal{P}$ determines that the generated text sequence $\hat{T}$ should not be spoken, we discard $\hat{T}$ and wait for the next chunk of input speech to be retrieved from the input sequence S. This iterative process continues until the entire input sequence S has been processed.

[8] https://drive.google.com/file/d/1Ht_SYswyzZfZARlmE3y8PLXx2xaUKRzH/view?usp=share_link

To avoid unnecessary execution dependencies and minimize output latency, we have implemented the Simultaneous speech to Text (SimulST) and Text-to-Speech (TTS) models on separate threads. Furthermore, when handling live microphone input, we have assigned a separate thread to the recorder that operates in the background during processing. This threading approach enables parallel execution of the SimulST, TTS, and recording tasks, optimizing the system's performance.

Our pipeline is designed in a modular fashion, allowing users to easily prototype and experiment with different policies for determining when to speak the generated output utterances. This modular approach empowers users to observe the effects of different policies on the latency and accuracy of the system's output, facilitating iterative refinement and experimentation in the development of SimulS2ST systems.

The consensus mechanism plays a crucial role in our pipeline, as it feeds the newly generated tokens to the Text-to-Speech (TTS) system for speech synthesis. To minimize pipeline latency, we have parallelized all stages of the pipeline. This means that the recorder, the speech to Text (ST) system, and the consensus mechanism operate concurrently without waiting for each other, enabling maximum audio throughput.

By leveraging parallelization, the recorder is not blocked by the ST system, and the ST system does not wait for the consensus mechanism. This design allows for efficient and uninterrupted processing of speech input, leading to reduced latency in the system overall. This parallelized approach ensures smooth and efficient operation of the pipeline, resulting in faster and more responsive speech to speech translation in real-time scenarios.

5.4. System Evaluation

We conduct a comprehensive evaluation of our system, focusing on translation into English from four typologically diverse languages, namely Japanese, Spanish, Russian, and Arabic. To build our offline ST model, we utilize OpenAI's Whisper (Radford et al., 2022), while for our TTS model, we query the ElevenLabs API (ElevenLabs, 2023). Our evaluations are performed on a single NVIDIA RTX 2080 GPU, and we report metrics based on a filtered subset of 75 examples from the dev set of

CoVoST2 (Wang et al., 2020a), specifically those examples that are at least 6 seconds in length. By using this approach, we ensure that our evaluations are conducted on a diverse range of languages and speech segments, providing a robust assessment of our system's performance.

To measure the accuracy of our system, we calculate the BLEU score between the spoken transcript T and the reference using the SacreBLEU package (Post, 2018). However, we choose not to use ASR BLEU on the output speech due to the potential lack of precision caused by cascading ASR errors. By utilizing the SacreBLEU package and focusing on the spoken transcript and reference, we aim to obtain reliable and meaningful accuracy metrics that reflect the quality of our system's translations without introducing potential errors from cascading ASR output.

To measure the latency of our system, we utilize the Computation-Aware Average Lagging (AL_{CA}) metric, which is an adaptation of the Average Lagging metric proposed by Ma et al. (2020c). The AL_{CA} metric takes into account the computational cost associated with generating an output, which is particularly relevant for our system as we also incur computational costs when speaking the output and context switching between threads. By incorporating the computational cost into our latency metric, we aim to obtain a more comprehensive and accurate assessment of the overall system performance, considering both translation quality and computational efficiency.

5.4.1. Policy Comparisons

We have compared four different policies in our system for determining when to speak the outputs. The first two policies serve as baseline approaches: the **Greedy Policy**[9], where $\mathcal{P}(\hat{T}) = \text{True}$, and the **Offline Policy**, where $\mathcal{P}(\hat{T}) = \text{False}$. These policies represent two extremes of the trade-off between translation quality and latency, allowing us to assess the upper and lower bounds of our system's performance in terms of these metrics.

Following the baseline policies, we have the **Confidence-Aware Policy (CAP)**, which involves returning true if the average probability of the sequence generated by the ST model (i.e., the confidence of the model) exceeds a certain threshold γ. During our testing, we observed that

[9]Equivalent to the wait-k policy in the SimulST literature (Ma et al., 2020c)

Window Size (t)	Japanese → English 1s	Japanese → English 2s	Spanish → English 1s	Spanish → English 2s
CAP ($\gamma = 0.9$)	20.7 (7.2)	21.1 (8.6)	38.3 (6.4)	41.5 (7.6)
CAP ($\gamma = 0.5$)	15.1 (4.9)	**18.8 (6.7)**	**25.8 (3.5)**	28.2 (5.4)
CP ($\alpha = 0.75$)	17.2 (6.8)	21.3 (9.6)	31.1 (4.6)	41.4 (7.9)
CP ($\alpha = 0.5$)	**15.3 (4.8)**	20.4 (8.8)	25.3 (4.1)	**35.6 (6.1)**
Greedy (wait-k)	5.7 (3.2)	10.0 (4.9)	8.6 (2.5)	15.2 (4.1)
Offline Policy	21.9 (9.5)	21.9 (9.5)	42.9 (9.6)	42.9 (9.6)

Window Size (t)	Russian → English 1s	Russian → English 2s	Arabic → English 1s	Arabic → English 2s
CAP ($\gamma = 0.9$)	36.6 (8.1)	37.4 (9.0)	18.8 (6.9)	19.5 (7.6)
CAP ($\gamma = 0.5$)	28.0 (4.1)	**31.8 (5.1)**	11.7 (4.1)	**13.7 (4.7)**
CP ($\alpha = 0.75$)	**27.2 (4.0)**	37.0 (8.5)	16.4 (6.2)	19.5 (8.7)
CP ($\alpha = 0.5$)	21.6 (3.3)	31.3 (6.2)	**11.0 (3.7)**	19.0 (7.3)
Greedy (wait-k)	12.8 (2.4)	17.3 (3.0)	3.8 (1.8)	5.9 (3.5)
Offline Policy	36.6 (10.0)	36.6 (10.0)	19.7 (9.1)	19.7 (9.1)

Table 5.1: Performance scores (BLEU, Average Lagging) for four policies (Offline, Greedy, Confidence-Aware (CAP), and Consensus (CP)) in SimulS2ST using Whisper Medium (769M params) (Radford et al., 2022). Optimal latency-quality trade-off shown in bold. Languages structurally similar to English exhibit better trade-off. Spanish and Russian BLEU scores approach Offline levels with minimal latency increase over Greedy policy.

using the Whisper field `no_speech_prob` yielded better empirical results compared to the returned `avg_logprob`, and thus we utilize it in our evaluation process.

Lastly, we have the **Consensus Policy (CP)**, which returns true only when the ratio of length to edit distance between the current transcript $\hat{T}_{k+1}$ and the previous transcript $\hat{T}_k$ falls below a certain threshold α. We run both the Consensus Policy and the Confidence-Aware Policy for two different parameter settings, showcasing the adaptability of our system to different latency regimes.

5.5. Results

We have developed a system that combines both translation and text-to-speech (TTS) modules to enable simultaneous speech to speech translation. With this technology, we are able to translate spoken words in the source language to text in the target language in real time, and then convert that text to speech in the target language, all in one seamless process.

We present our evaluation results in Table 5.1. Notably, we observe that languages that share greater structural similarity with English tend to achieve lower latency on average across all policies. Specifically, we observe latency reductions in most policies for Spanish, while only in a few cases for Japanese. This highlights the significance of fine-tuning model input parameters on a per-language basis, emphasizing the need for language-specific optimization approaches.

Moreover, our evaluation demonstrates that for certain languages, our system achieves significant improvements in both translation quality and latency compared to the Greedy baseline, while maintaining a comparable translation quality to the offline setting. For instance, when translating Spanish to English using the Confidence-Aware Policy with a threshold of $\gamma = 0.5$, we achieve a remarkable 17-point increase in BLEU score, with only a minor sacrifice of 1 second in average lagging. Similarly, in the case of Russian, employing the Consensus Policy with a threshold of $\alpha = 0.75$ results in a nearly 15-point increase in BLEU score, with a modest increase of 1.6 seconds in latency. These high BLEU scores indicate the promising practical utility of our system in real-world scenarios.

To showcase the capabilities of this powerful system, we have included a recording of a live demon-

stration on this link[10]. This recording highlights the simultaneous translation process in action, showcasing the speed and accuracy of the technology as it seamlessly translates spoken words from one language to another.

[10]https://drive.google.com/file/d/1Ym6mB9XDPiN7KuzmpkMbKe9PVom6ULGs/view?usp=sharing

CHAPTER 6

Conclusions

Section 4.1 of our study described a preliminary experiment that explored the use of hidden units as intermediate representations for the speech to speech translation task. However, the results showed that these units were unable to capture input speaker speech characteristics, such as pauses, emotion, inflection, pitch, and accent. As such, we concluded that hidden units were unsuitable as a replacement for text in cascaded approaches aimed at solving the task of simultaneous speech to speech translation. In Section 4.2, we evaluated an end-to-end speech to speech translation model and found that it struggled with degeneration when not using teacher forcing. Moreover, it was unable to produce unintelligible outputs, which made us realize that the end-to-end setting was inappropriate for the real-time translation use case. In Section 4.3, we found that monotonic attention is better suited for the task of simultaneous speech to speech translation due to the lack of the whole audio clip while calculating attention.

Although SimulS2ST (Simultaneous speech to speech Translation) is a field that is currently being actively researched, it has already proven to be highly practical in enhancing communication. To further support the development and benchmarking of SimulS2ST systems, we have developed a customizable baseline system that allows users to dynamically adjust policy parameters, thereby directly influencing the trade-off between latency and translation quality. Through our evaluations, we have observed that these policy parameters achieve comparable accuracy to offline models, while significantly improving latency performance. We believe that our system will be valuable for both industry professionals and researchers in prototyping and advancing future SimulS2ST systems, and we hope that our contributions will contribute to the ongoing progress in this field.

6.1. Future Work

To further investigate the nature of hidden units, one potential direction of study could be to train the Hidden Unit encoder network with inflected speech to see if it can learn these features. In this case, the training dataset would need to be synthetically generated by varying speech inflection on

possible tokens of interest.

Another promising direction includes providing a detailed comparison of end-to-end simultaneous speech to speech translation with the one that uses text as an intermediate representation. This would help to further understand the trade-offs between these two approaches and identify their respective strengths and weaknesses.

BIBLIOGRAPHY

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. Common voice: A massively-multilingual speech corpus, 2020.

Naveen Arivazhagan, Colin Cherry, Wolfgang Macherey, Chung-Cheng Chiu, Semih Yavuz, Ruoming Pang, Wei Li, and Colin Raffel. Monotonic infinite lookback attention for simultaneous machine translation, 2019. URL https://arxiv.org/abs/1906.05218.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations, 2020a.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations, 2020b.

Alexei Baevski, Wei-Ning Hsu, Alexis CONNEAU, and Michael Auli. Unsupervised speech recognition. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 27826–27839. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/ea159dc9788ffac311592613b7f71fbb-Paper.pdf.

William Chan, Daniel Park, Chris Lee, Yu Zhang, Quoc Le, and Mohammad Norouzi. Speechstew: Simply mix all available speech recognition data to train one large neural network, 2021.

Guoguo Chen, Shuzhou Chai, Guanbo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, Mingjie Jin, Sanjeev Khudanpur, Shinji Watanabe, Shuaijiang Zhao, Wei Zou, Xiangang Li, Xuchen Yao, Yongqing Wang, Yujun Wang, Zhao You, and Zhiyong Yan. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio, 2021a.

Junkun Chen, Mingbo Ma, Renjie Zheng, and Liang Huang. Direct simultaneous speech-to-text translation assisted by synchronized streaming asr, 2021b. URL https://arxiv.org/abs/2106.06636.

Chung-Cheng Chiu and Colin Raffel. Monotonic chunkwise attention, 2017. URL https://arxiv.org/abs/1712.05382.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. MuST-C: a Multilingual Speech Translation Corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies, Volume 1 (Long and Short Papers), pages 2012–2017, Minneapolis, Minnesota, June 2019a. Association for Computational Linguistics. doi: 10.18653/v1/N19-1202. URL https://aclanthology.org/N19-1202.

Mattia Antonino Di Gangi, Matteo Negri, Roldano Cattoni, Roberto Dessi, and Marco Turchi. Enhancing transformer for end-to-end speech-to-text translation. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 21–31, Dublin, Ireland, August 2019b. European Association for Machine Translation. URL https://aclanthology.org/W19-6603.

ElevenLabs. Elevenlabs API. https://api.elevenlabs.io/docs, 2023. Accessed: 2023-04-10.

Daniel Galvez, Greg Diamos, Juan Ciro, Juan Felipe Cerón, Keith Achorn, Anjali Gopi, David Kanter, Maximilian Lam, Mark Mazumder, and Vijay Janapa Reddi. The people's speech: A large-scale diverse english speech recognition dataset for commercial usage, 2021.

Dapeng Hong. Attention-based recurrent neural networks for question answering. 2017.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units, 2021. URL https://arxiv.org/abs/2106.07447.

Hirofumi Inaguma, Sravya Popuri, Ilia Kulikov, Peng-Jen Chen, Changhan Wang, Yu-An Chung, Yun Tang, Ann Lee, Shinji Watanabe, and Juan Pino. Unity: Two-pass direct speech-to-speech translation with discrete units, 2022.

Ye Jia, Ron J. Weiss, Fadi Biadsy, Wolfgang Macherey, Melvin Johnson, Zhifeng Chen, and Yonghui Wu. Direct speech-to-speech translation with a sequence-to-sequence model, 2019. URL https://arxiv.org/abs/1904.06037.

Ye Jia, Michelle Tadmor Ramanovich, Tal Remez, and Roi Pomerantz. Translatotron 2: Robust direct speech-to-speech translation. *arXiv preprint arXiv:2107.08661*, 2021.

Ye Jia, Michelle Tadmor Ramanovich, Tal Remez, and Roi Pomerantz. Translatotron 2: High-quality direct speech-to-speech translation with voice preservation. In *International Conference on Machine Learning*, pages 10120–10134. PMLR, 2022a.

Ye Jia, Michelle Tadmor Ramanovich, Quan Wang, and Heiga Zen. CVSS corpus and massively multilingual speech-to-speech translation. In *Proceedings of Language Resources and Evaluation Conference (LREC)*, pages 6691–6703, 2022b.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in Neural Information Processing Systems*, 33:17022–17033, 2020.

Ann Lee, Peng-Jen Chen, Changhan Wang, Jiatao Gu, Sravya Popuri, Xutai Ma, Adam Polyak,

Yossi Adi, Qing He, Yun Tang, Juan Pino, and Wei-Ning Hsu. Direct speech-to-speech translation with discrete units, 2021a. URL https://arxiv.org/abs/2107.05604.

Ann Lee, Hongyu Gong, Paul-Ambroise Duquenne, Holger Schwenk, Peng-Jen Chen, Changhan Wang, Sravya Popuri, Yossi Adi, Juan Pino, Jiatao Gu, and Wei-Ning Hsu. Textless speech-to-speech translation on real data, 2021b. URL https://arxiv.org/abs/2112.08352.

Ann Lee, Hongyu Gong, Paul-Ambroise Duquenne, Holger Schwenk, Peng-Jen Chen, Changhan Wang, Sravya Popuri, Yossi Adi, Juan Pino, Jiatao Gu, and Wei-Ning Hsu. Textless speech-to-speech translation on real data, 2022.

Bing Liu and Ian Lane. Attention-based recurrent neural network models for joint intent detection and slot filling, 2016.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

Xutai Ma, Juan Pino, James Cross, Liezl Puzon, and Jiatao Gu. Monotonic multihead attention, 2019. URL https://arxiv.org/abs/1909.12406.

Xutai Ma, Mohammad Javad Dousti, Changhan Wang, Jiatao Gu, and Juan Pino. Simuleval: An evaluation toolkit for simultaneous translation, 2020a.

Xutai Ma, Juan Pino, and Philipp Koehn. SimulMT to SimulST: Adapting simultaneous text translation to end-to-end simultaneous speech translation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 582–587, Suzhou, China, December 2020b. Association for Computational Linguistics. URL https://aclanthology.org/2020.aacl-main.58.

Xutai Ma, Juan Pino, and Philipp Koehn. Simulmt to simulst: Adapting simultaneous text translation to end-to-end simultaneous speech translation. *arXiv preprint arXiv:2011.02048*, 2020c.

Stephen Merity. Single headed attention rnn: Stop thinking with your head, 2019.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015a. doi: 10.1109/ICASSP.2015.7178964.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015b. doi: 10.1109/ICASSP.2015.7178964.

Sara Papi, Marco Gaido, Matteo Negri, and Marco Turchi. Does simultaneous speech translation

need simultaneous models? *arXiv preprint arXiv:2204.03783*, 2022.

Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-6319.

Matt Post, Gaurav Kumar, Adam Lopez, Damianos Karakos, Chris Callison-Burch, and Sanjeev Khudanpur. Improved speech-to-text translation with the fisher and callhome Spanish-English speech translation corpus. In *Proceedings of the 10th International Workshop on Spoken Language Translation: Papers*, Heidelberg, Germany, December 5-6 2013. URL https://aclanthology.org/2013.iwslt-papers.14.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision, 2022.

Colin Raffel, Minh-Thang Luong, Peter J. Liu, Ron J. Weiss, and Douglas Eck. Online and linear-time attention by enforcing monotonic alignments, 2017. URL https://arxiv.org/abs/1704.00784.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Changhan Wang, Anne Wu, and Juan Pino. Covost 2 and massively multilingual speech-to-text translation, 2020a.

Changhan Wang, Anne Wu, and Juan Pino. Covost 2 and massively multilingual speech-to-text translation, 2020b.

Changhan Wang, Morgane Rivière, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation, 2021.

Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding, 2020.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech, 2019.

Yu Zhang, Daniel S. Park, Wei Han, James Qin, Anmol Gulati, Joel Shor, Aren Jansen, Yuanzhong Xu, Yanping Huang, Shibo Wang, Zongwei Zhou, Bo Li, Min Ma, William Chan, Jiahui Yu, Yongqiang Wang, Liangliang Cao, Khe Chai Sim, Bhuvana Ramabhadran, Tara N. Sainath, Francoise Beaufays, Zhifeng Chen, Quoc V. Le, Chung-Cheng Chiu, Ruoming Pang, and Yonghui

Wu. BigSSL: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1519–1532, oct 2022. doi: 10.1109/jstsp.2022.3182537. URL https://doi.org/10.1109%2Fjstsp.2022.3182537.