Sign Language Translation with Transformers

By

[REDACTED]

A thesis submitted to
[redacted]
for the degree of
BACHELOR OF SCIENCE

logo.png

[redacted]
April 2020

ABSTRACT

This thesis explores a Neural Machine Translation (NMT) based approach towards automatic Sign Language Translation (SLT). The communication barrier between the Deaf community and hearing people prevents those with hearing loss to easily interact with a predominantly hearing society, and presents numerous challenges in their daily lives. SLT poses an interesting challenge and the provision of such a system can facilitate communication between the Deaf and hearing.

We begin by addressing the challenges of sign language processing and examine the different steps in SLT. Existing SLT systems first use a Sign Language Recognition (SLR) system to extract sign language glosses from videos. Then, a translation system generates spoken language translations from the sign language glosses. Our survey of previous efforts in SLT shows that though SLT has gathered interest recently, little study has been performed on the translation system. This thesis focuses on enhancing translation from sign language glosses to spoken language.

The recent success of Transformers for NMT between spoken languages inspires us to adopt this architecture. We study Transformers for SLT in various setups, including techniques from spoken language processing that have not yet been applied to sign language. Our experiments on RWTH-PHOENIX-Weather 2014T, a challenging SLT benchmark dataset of German sign language, and ASLG-PC12, a dataset involving American Sign Language (ASL), confirm our hypothesis that Transformers achieve better SLT results than previous RNN-based architectures.

Our methodology improves on the current state-of-the-art by over 5 and 7 points respectively in BLEU-4 score on ground truth glosses and predict glosses of RWTH-PHOENIX-Weather 2014T. On ASLG-PC12, we report an improvement of over 16 points. Our findings also demonstrate that end-to-end translation from videos provides even better results than translation of ground truth glosses. This shows potential for further improvement in SLT by either jointly training the SLR and translation systems or by revising the annotation system of sign language videos.

DEDICATION

Dedicated to the brave healthcare workers and other essential workers of the world who are risking their lives on the front lines of the battle against the COVID-19 pandemic at the time of this thesis.

ACKNOWLEDGMENTS

Undertaking this Bachelor's thesis was a truly rewarding experience that only encourages me to continue pursuing research in Natural Language Processing, and I cannot thank enough all those who made this possible.

I would first like to thank Prof. [redacted] for his unrivalled support and encouragement during my thesis. I have been extremely lucky to have an exemplary supervisor who was always approachable and available, whose confidence in my research inspired me to tackle ambitious challenges, who offered me the right balance between guidance and autonomy.

I also thank the [redacted] group at [redacted] for making me feel belonging to the lab from day 1, for their interest in my work and our fun discussions over lunch, especially [redacted] who kindly welcomed me and provided the smoothest administrative support for my internship. A special thanks to [redacted] who generously shared his time and expertise for this project, for all the insightful discussions over coffee, whose technical help was indispensable and whose suggestions greatly contributed to this thesis.

I would also like to mention Dr. [redacted] from the [redacted] lab in [redacted] who kindly shared data and details about the Spatial-Temporal Multi-Cue network with me.

Finally, I am grateful to my parents who made sure I was in the best conditions while I worked from home during the last weeks of my thesis, my SSH connection that allowed me to continue accessing the GPU during confinement, and my friends, my love, my dog, my piano and Buddha's teachings for their unconditional support and help in coping with social distancing.

Table of Contents

Appendices

List of Figures

List of Tables

Chapter One

Introduction

Blindness cuts us off from things, but deafness cuts us off from people.

– Helen Keller, American deaf-blind educator and activist

1.1 Motivation

Communication holds a central position in our daily lives and social interactions. Yet, in a predominantly hearing society, people with hearing loss are often deprived of effective communication. Although Deaf communities have developed sign language as means of communication, the use of sign language often remains restricted to these communities.

Lip-reading is an imperfect solution for deaf people to understand spoken language because it relies on their good knowledge of the spoken language and the speaker must talk slowly with good articulation. Even then, it is prone to interpretation errors. Emulating spoken language for the deaf may not be feasible depending on the nature of their deafness. Sign language is a complex language that takes years of practice to master and its meaning cannot be easily guessed by non-practitioners. Sign language interpreters are not always readily available, can become costly and may compromise the individual's privacy.

While advancements have been made to better accommodate deaf people, such as video captioning and increased use of online text-based communication, the Deaf community still face issues of social isolation and miscommunication daily [13, 49, 56, 64]. Moreover, many of these methods rely on the Deaf to read spoken language text. Recently, machine translation enabled people to better understand a foreign spoken language. This thesis is motivated to perform machine translation from sign to spoken language[1] to enable communication between the Deaf and hearing while respecting each person's preferred language.

1.2 Towards Machine Translation

Machine translation of sign language to spoken language usually involves two steps. From a video, an automatic sign language recognition system first transcribes the signs into text form. Since sign language varies in grammar and structure from spoken language, a second step is needed to translate the recognized sequence of signs into spoken language.

Recent works [43, 69] improves the first step, but there has been no work improving the second step to this date. This thesis aims to address this research gap by leveraging recent successes in Neural Machine Translation to enhance the translation system. We namely focus our study on the Transformer [63], a state-of-the-art NMT model.

1.3 Organization

The remaining chapters in this thesis are organized as follows: **Chapter Two** discusses the challenges of sign language processing and the sub-tasks involved in sign language translation (SLT); **Chapter Three** provides a brief survey of related work in SLT; **Chapter Four** introduces the architecture of the main models we use; **Chapter Five** presents the datasets for SLT; **Chapter Six** details our experimental work and discusses the results; and **Chapter Seven** summarizes achievements of this thesis and proposes future research directions.

[1] The term "spoken language" is used in this thesis to distinguish them from sign language and does not refer specifically to their oral version. We mainly work with spoken language in their written form.

Chapter Two

Background

2.1 Challenges of Sign Language Processing

Despite considerable advancements made in machine translation between spoken languages, sign language processing falls behind for many reasons. Having only emerged in the 1960s with [60], research in sign language is still in its infancy and only 41 countries today legally recognize sign language as an official language. Moreover, there is not one universal sign language. Ethnologue[1] lists 144 different sign languages where most are not mutually intelligible with each other. For this reason, SLT would not only improve communication between the Deaf and hearing, but also between different Deaf communities around the globe.

Unlike spoken language, sign language is a multidimensional form of communication that relies on both manual and non-manual cues which presents additional computer vision challenges [2]. Sign language relies not only on hand movements but also other means of body language such as facial expressions, body posture, head and upper body movements and word mouthings. These cues may occur simultaneously whereas spoken language follow a simpler linear pattern where only a single word is processed at a time. Sign languages also make use of 3D space in a unique way to locate concepts[2]. For example, sign language users make use of space to describe topographical relationships. The signer can also introduce a

[1] https://www.ethnologue.com

[2] https://bsl.surrey.ac.uk/principles/g-signing-space

person to the conversation by fingerspelling their name once and placing them in the signing space, then refer back to person by simply pointing to the assigned space. Signs also vary in both space and time, where two sequences of the same signs may be performed at different speeds, with gestures of different magnitudes or at different positions from the camera, and the number of video frames associated to a single sign is not fixed either.

In addition, sign languages generally developed independently of spoken language and often do not share the grammar of their spoken counterparts. For instance, the syntax of ASL shares more with spoken Japanese than English [42]. For this reason, sign language processing is not only a difficult multi-modal visual recognition problem, but also a significant linguistic challenge. An effective machine translation system must be able to process the rich linguistic features and unique grammar of each sign language. However, it is difficult to obtain suitable sign language data to effectively train a machine translation system. There is no standard on how to annotate sign language data. Compared to spoken language text, data collection of sign language requires additional effort and time, and due to the relative smaller volume of existing sign language research, available datasets for sign language processing are often very limited in size, vocabulary and/or quality [41, 55, 59].

2.2 Sign Language Glossing

Machine translation uses an intermediate textual representation of sign language to perform translation. Though several transcription schemes for sign language have been proposed [51, 60], none of them have been formally adopted and sign language lacks a standard writing system. Because of the multi-modality and non-linearity of sign language, it is not straightforward to transpose the full meaning of sign language into written form.

One way to represent sign language in written form is by transcribing the meaning of the signs, which is what glossing aims to do. Glossing corresponds to transcribing sign language word-for-word by means of another written language, and can be more or less detailed depending on the transcriber. Glosses differ from translation as they merely indicate what each part in a sign language sentence mean, but does not form an appropriate sentence in the spoken language. In general, sign language glosses are annotated by hand by sign language experts, and more recently by continuous sign language recognition (CSLR) systems (cf. Section 2.3). Gloss annotations are well-suited for machine translation because the transcriber can easily control the level of linguistic detail needed in the annotations, and sign variants can be denoted by the same gloss which removes the need for lemmatization[3].

However, while various sign language corpus projects have provided different guidelines for gloss annotation [14, 15, 25], there is no single standard agreed on [26, 54] which hinders the easy exchange of data between projects and consistency between different sign language corpora. Gloss annotations also remain an imprecise representation of sign language and can inevitably lead to an information bottleneck when representing the multi-channel sign language by a single-dimensional stream of glosses (cf. Chapter 6).

2.3 Sign Language Recognition

Over the last decade, some progress has been made in sign language recognition (SLR) as well as CSLR for various sign languages. SLR consists of identifying isolated single signs from videos, and CSLR is a relatively more challenging task that consists of identifying a sequence of running glosses from a given video. Most SLR and CSLR systems make predictions on RGB video data [3, 17], as is the case with the CSLR system we use in our experiments. Nevertheless, our translation system may be used with any other SLR systems that recognize glosses sequences as well, such as those using gloves or accelerometers [53, 67]. We discuss related work in SLR in the next chapter.

[3] Lemmatization consists of grouping the derived forms of a word to its dictionary form and is useful during natural language processing to analyze these forms as a single item.

2.4 Sign Language Translation

SLT differs from SLR as the latter merely detects a sequence of signs without taking into account the linguistic structures and grammar unique to sign language. As illustrated in Figure 2.1, the SLT system takes CSLR as a first step to detect a sequence of glosses from the input video. Then, an additional step translates the detected glosses into a valid sentence in the target language. SLT is a novel problem and a difficult task compared to other translation problems because it involves two steps to extract meaningful features from a video of a multi-cue language accurately, then generate translations from an intermediate gloss representation, instead of performing translation on the source language directly.

While SLR is monotonous, that is each gloss is recognized in the order it appears in the input video, the word orders of the source and target sequences in machine translation often differ and word forms in the target depend on its context. SLT therefore presents a significant complexity in computational linguistics as the system must compute the probability of different word orders and grammatical forms conditionally to the relationship between words within sequences. Our aim is to tackle this additional challenge and enhance the translation system by using natural language processing techniques and state-of-the-art architectures, namely the Transformer (cf. Chapter 4).

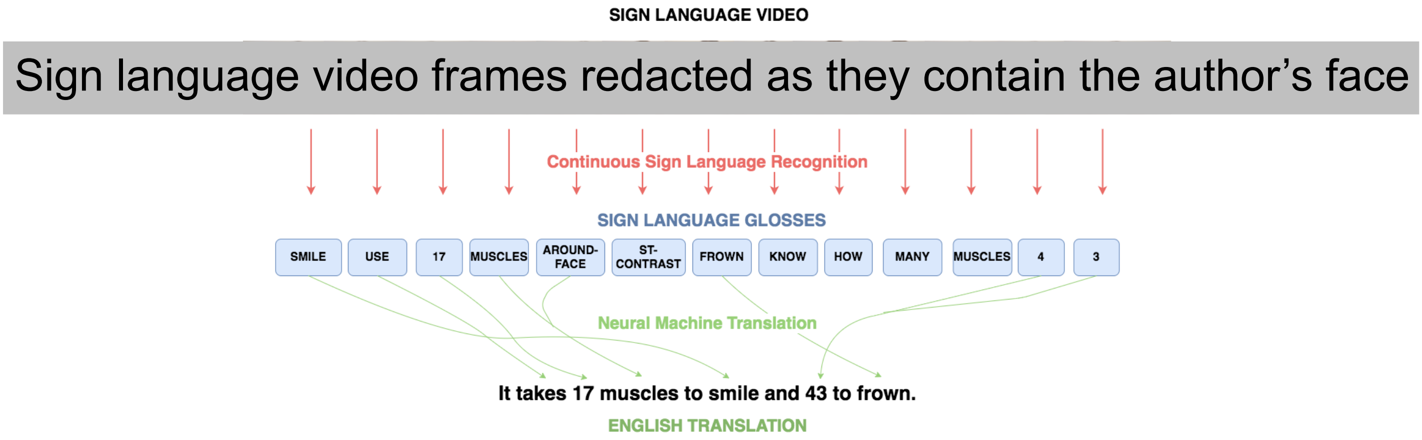
SIGN LANGUAGE VIDEO
Sign language video frames redacted as they contain the author’s face
Continuous Sign Language Recognition
SIGN LANGUAGE GLOSSES
SMILE
USE
17
MUSCLES
AROUND-FACE
ST-CONTRAST
FROWN
KNOW
HOW
MANY
MUSCLES
4
3
Neural Machine Translation
It takes 17 muscles to smile and 43 to frown.
ENGLISH TRANSLATION

Figure 2.1 Sign language translation. This task consists of successively performing CSLR and NMT. Glosses obtained from [36].

Chapter Three

Related Work

3.1 Sign Language Recogition

The first approaches for SLR rely on hand-crafted features [7, 12, 62, 65, 66] and use Hidden Markov Models [18, 19, 33, 57, 58] or Dynamic Time Warping [37] to model sequential dependencies. More recently, 2D convolutional neural networks (2D-CNN) [16] and 3D convolutional neural networks (3D-CNN) [40] have shown to be effective on modelling spatio-temporal representations from sign language videos.

Most existing works on CSLR divide the task into three sub-tasks: alignment learning, single-gloss SLR, and sequence construction [34, 68] while others perform the task in an end-to-end fashion using deep learning architectures [8, 23, 24]. Figure 3.1 gives an overview of a CSLR system using Bayes' decision rule to find the sequence of words that best fit the trained word models and the language model. Works in SLR and CSLR, however, focus mainly on the visual recognition task and does not take into account the underlying grammatical and linguistic features of sign language.

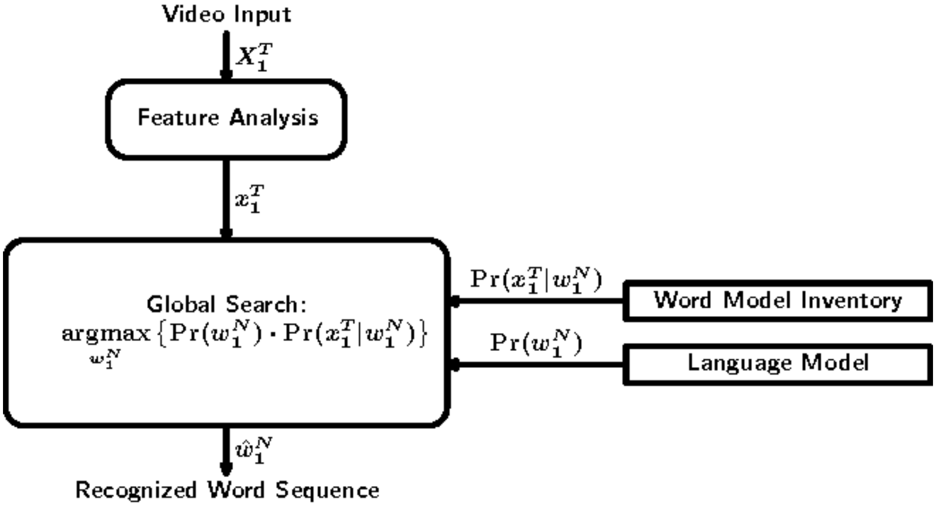

Figure 3.1 Overview of a CSLR system. Figure from [18].

3.2 Sign Language Translation

SLT is formalized for the first time in [9] where they introduce the RWTH-PHOENIX-Weather 2014T dataset and jointly use a 2D-CNN model to extract gloss-level features from video frames, and RNN-based sequence-to-sequence models to perform translation on German sign language. Subsequent works have been published using this dataset [43, 69], but all of them focus only on improving the CSLR component in SLT. Despite recent advancements in the field of NMT, no study has been made so far seeking to improve the baseline on translating glosses to spoken language text.

Similar work has been done for Korean sign language by [32] where they estimate human keypoints to extract glosses, then use RNN-based sequence-to-sequence models for translation. [1] uses RNN-based sequence-to-sequence models to directly translate ASL glosses from the ASLG-PC12 dataset [44] and does not involve sign language videos.

3.3 Neural Machine Translation

Neural Machine Translation (NMT) employs neural networks to perform automated text translation. NMT approaches typically use an encoder-decoder architecture, also known as sequence-to-sequence (seq2seq) models. Figure 3.2 shows a seq2seq model with attention where the encoder reads the input sentence one word at a time to produce a hidden representation. Then, the decoder produces words in the target sentence until the end of sentence (<EOS>). At each step, the decoder uses the previous predicted word and attention computed on the full input sentence to decide which input words are the most relevant.

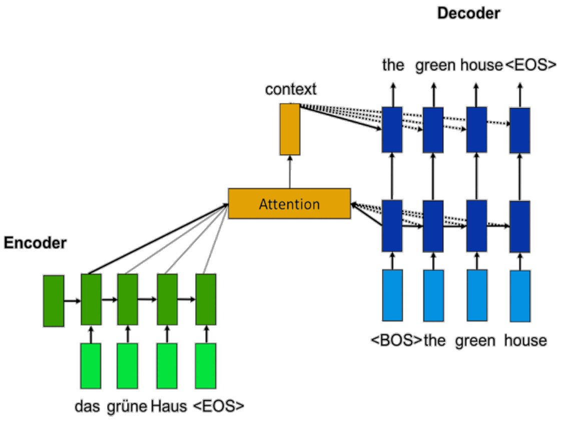

Figure 3.2 Seq2seq model with attention. Figure from [22].

Earlier approaches use recurrent networks [4, 10, 28, 61] and convolutional networks [20, 27, 29] for the encoder and decoder. However, these seq2seq networks are unable to model long-term dependencies in large input sentences without causing an information bottleneck. To address this issue, more recent works use attention mechanisms introduced by [4] and later extended by [39]. Their attention function calculates context-dependent alignment scores between encoder and decoder hidden states. In other words, the attention mechanism assign scores to different hidden states and amplify states with higher scores that are more relevant to the word being translated while reducing states with lower scores. [63] introduces the Transformer architecture, a seq2seq model that relies on self-attention and does not implicate any recurrent networks, that obtains state-of-the-art results in NMT.

Chapter Four

Model architectures

4.1 Transformer

Transformer [63] is a seq2seq network that differs from previous models in its usage of self-attention layers instead of recurrent networks. Its architecture is illustrated in Figure 4.1. In this section we will briefly explain its architecture[1].

4.1.1 Overall architecture

Transformers are composed of an encoding component and a decoding component. Both the encoder and decoder stacks are composed of N identical layers. Each word in an input sentence is first embedded into a vector of size d_{model} before being passed on to the first layer of the encoder stack. All encoder layers receive a list of n vectors of size d_{model}. The output of the last encoder layer is then used by each decoder layer during its attention operation. At each step, the decoder stack is auto-regressive meaning it uses previously generated symbols as additional input when generating the next symbols.

[1] This blog provides great visualizations and explanation of Transformers in more detail http://jalammar.github.io/illustrated-transformer

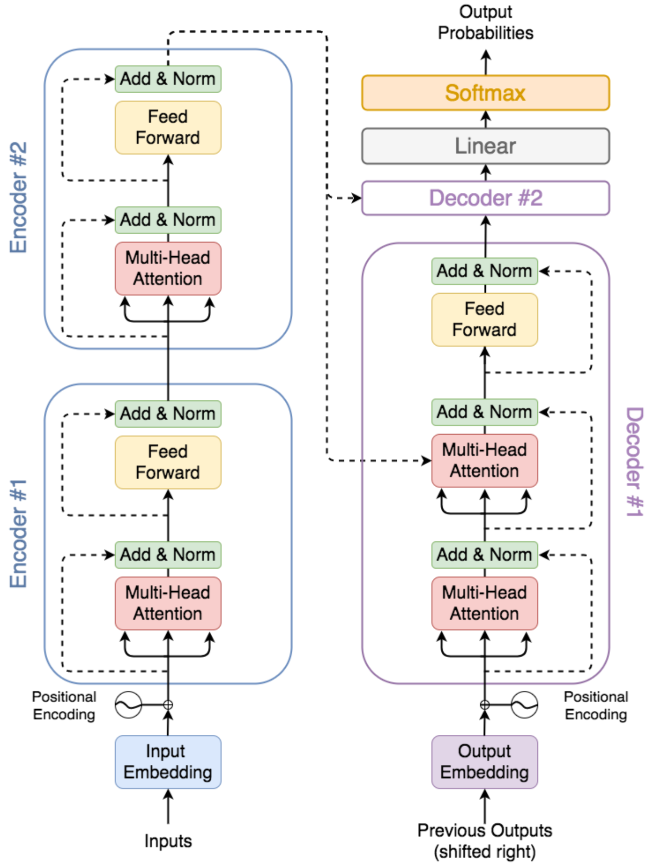

Figure 4.1 Architecture of a Transformer with two encoder-decoder layers.

4.1.2 Multi-head Attention

Intuitively, when the model processes each word, self-attention guides the model to look at other words in the input sentence for clues to obtain a better encoding of the word.

The attention mapping takes as input a set of n queries $Q \in R^{n \times d_k}$, keys $K \in R^{n \times d_k}$ and values $V \in R^{n \times d_v}$ to produce an output $O \in R^{n \times d_v}$:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

In addition, multi-head attention employs h parallel attention layers, or heads, to obtain multiple representation subspaces and allow the model to focus on different positions.

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_h)W^O$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$ with $W_i^Q, W_i^K \in R^{d_{model} \times d_k}$,$W_i^V \in R^{d_{model} \times d_v}$ and $W_i^O \in R^{hd_v \times d_{model}}$.

4.1.3 Encoder

In the encoder, each layer is composed of two sub-layers: a multi-head attention mechanism and a position-wise, fully-connected feed-forward layer. Around each sub-layer is also a residual connection followed by layer normalization. That is, the output of each sub-layer is $LayerNorm(x + Sublayer(x))$ where x is the encoder input and $Sublayer$ denotes the function applied by the sub-layer itself.

4.1.4 Decoder

In addition to the same two sub-layers as in a encoder layer, each decoder layer has an additional "Encoder-Decoder attention" sub-layer. This sub-layer has the same mechanism as multi-head attention, except that it uses queries from the layer below it, and keys and values from the output of the encoder stack (Figure 4.1). At each time step, the decoder stack outputs a symbol from the output sentence, which is then fed to the first decoder layer in the next time step, until the symbol indicating the end of the sentence is reached. The self-attention layers in the decoder also mask future positions, by setting them to $-inf$ for example, so that the predictions for the i-th symbol can only depend on known outputs at positions less than i.

4.1.5 Positional Encoding

Since the Transformer contains no recurrence or convolution, the network with self-attention has no notion of the word order in a sentence. To address this, positional encoding is summed with the input embeddings at the bottoms of the encoder and decoder stacks. Positional encoding adds information about the relative position of symbols and allows the model to make use of the order of the words. The positional encoding vector $p = (p_1, p_2, ..., p_m)$ with $p_j \in R^f$ is obtained using sine and cosine functions of different frequencies:

$$p_{2i} = \sin(pos/10000^{2i/d_{model}})$$
$$p_{2i+1} = \cos(pos/10000^{2i/d_{model}})$$

where *pos* is the position of the symbol in its sentence.

4.2 Spatial-temporal multi-cue network

For our experiments on end-to-end SLT from sign language videos to text translations, we use spatial-temporal multi-cue (STMC) network for CSLR proposed by [69] in conjunction with Transformers. The STMC network obtains word error rate 19.6 and 21.0 on the dev and test sets of RWTH-PHOENIX-Weather 2014T which is the current state-of-the-art in CSLR on this dataset. We will give a brief overview of this framework and refer to the original paper for more details.

Figure 4.2 shows the three key modules used by the STMC model. First, a spatial multi-cue (SMC) module decomposes the input video into spatial features of multiple visual cues including full-frame, hand, face and pose. Strips of different colors represent sequences of features of different cues. Then, a temporal multi-cue (TMC) module calculates temporal correlations within and between cues at different time steps. The intra-cue and inter-cue features are each analyzed by Bi-directional Long Short-Term Memory (BLSTM) [61] and Connectionist Temporal Classification (CTC) [21] units for sequence learning and inference. The three modules are trained in an end-to-end fashion in order to analyze multi-cue data.

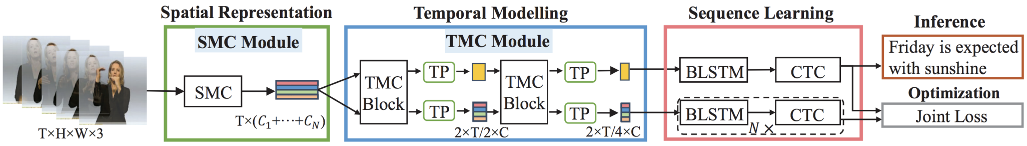

Figure 4.2 Overview of the STMC network. Figure from [69].

Chapter Five

Datasets

5.1 RWTH-PHOENIX-Weather 2014T

To evaluate and compare the performance of our model to existing works, we use the RWTH-PHOENIX-Weather 2014T dataset introduced by [9]. The data is extracted from news and weather forecast airings of the German tv station PHOENIX, and to our knowledge it is currently the only publicly available dataset with both gloss level annotations and spoken language translations for sign language videos that is of sufficient size and challenge for deep learning.

This dataset consists of a parallel corpus of German sign language videos from 9 different signers, gloss-level annotations with a vocabulary of 1,066 different glosses and translations into German spoken language with a vocabulary of 2,887 different words.

5.2 ASLG-PC12

As can be seen in Table 5.1, the RWTH-PHOENIX-Weather 2014T dataset only has 7,096 training pairs, while deep learning models, notably Transformers, achieve better results on larger datasets [48]. We would also like to assess the performance of our model on a larger dataset and in a language most of us are familiar with. For this purpose, we conduct NMT

Table 5.1 Statistics of the RWTH-PHOENIX-Weather 2014T and ASLG-PC12 datasets. Out-of-vocabulary (OOV) words are those that appear in the development and testing sets, but not in the training set. Singletons are words that appear only once during training.

	German Sign Gloss			German			American Sign Gloss			English		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Phrases	7,096	519	642	7,096	519	642	82,709	4,000	1,000	82,709	4,000	1,000
Vocab.	1,066	393	411	2,887	951	1,001	15782	4,323	2,150	21,600	5,634	2,609
tot. words	67,781	3,745	4,257	99,081	6,820	7,816	862,046	41,030	10,503	975,942	46,637	11,953
tot. OOVs	–	19	22	–	57	60	–	255	83	–	369	99
singletons	337	–	–	1,077	–	–	6,133	–	–	8,542	–	–

of sign language glosses on the ASLG-PC12 corpus proposed in [44]. So far, SLT on this dataset has only been performed using RNN-based sequence-to-sequence attention networks in [1].

This corpus consists of 87,709 pairs of ASL gloss with a vocabulary of 15,782 different glosses and English sentences with a vocabulary of 21,601 different words. It is constructed from English data of the Project Gutenberg that has been transformed into ASL glosses following an automatic rule-based approach and validated by human experts. There are no explicit training, development and testing splits[1] published on the dataset so we created our own random splits for our experiments. The splits and our code are made publicly available[2] to encourage and underpin future research.

[1] Training data is used to fit the model. The model is evaluated on the development or validation data during training. The testing data is used once the model finished training to evaluate its performance on unseen data.

[2] https://github.com/[redacted]

Chapter Six

Experiments and Discussions

6.1 Experiment setup

All of our Transformer models are built using PyTorch [46] and the Open-NMT library [31] with a word embedding size of 512 and Transformer feed-forward layers of 2048 hidden units. For optimization, we use Adam [30] with 0.9 beta1 and 0.998 beta2, as well as Noam learning rate schedule, 0.1 dropout, and 0.1 label smoothing. Our experiments are run on NVIDIA Titan Xp GP102 12GB.

During training, the networks are evaluated on the dev set each half-epoch, and early stopping with patience 5 is used to halt training. Decoding is performed using beam search with a beam width of 5. During decoding, generated <unk> tokens for unknown words are also replaced by the source token having the highest attention weight[31]. This is useful when unknown symbols correspond to proper nouns that can be directly transposed between languages.

As our architecture varies highly from previous works on SLT, and especially since Transformers are highly sensitive to hyperparameter and architecture settings, we perform a series of experiments to find the optimal setup. We equally experiment with various techniques often used in classic NMT to SLT such as transfer learning, weight tying and ensembling to improve model performance.

Finding a suitable evaluation method is an ongoing challenge in NLP where natural language is often highly ambiguous. We use three metrics for automatic evaluation: Bi-Lingual Evaluation Understudy (BLEU, [45]), Recall-Oriented Understudy for Gisting Evaluation (ROUGE, [38]) and METEOR Automatic Machine Translation Evaluation System [5]. BLEU calculates the n-gram precision between the prediction and ground truth, and is currently the most widely used to evaluate NMT tasks. ROUGE is based entirely on recall and is more often used for text summarization evaluation, but we include ROUGE scores for consistency with existing literature on SLT. METEOR computes the harmonic mean of unigram precision and recall, and has shown to give better correlation to human judgement than BLEU on NMT evaluation. For BLEU, we report BLEU-1,2,3,4 scores to give better perspective of model performance on different levels and as ROUGE score we report the ROUGE-L F1 score. We also provide qualitative results in Appendix A and B.

Finding the optimal model setup consists of running several grid searches over different hyperparameters and configurations, where we exhaustively search through different combinations of model configurations. This is necessary as hyperparameters are not always independent of each other, as confirmed during our experiments. For the purpose of reporting the influence of different configurations to model behavior however, we present selected results usually varying one configuration at a time in the following sections. All reported results are averaged over 10 runs with different random seeds. The variance between experiments with different seeds is under 10^{-1} points on each score.

We organize our experiments into two groups:

1. Gloss2Text (G2T) in which we translate ground truth gloss annotations to simulate perfect tokenization, on both RWTH-PHOENIX-Weather 2014T and ASLG-PC12

2. Sign2Gloss2Text (S2G2T) in which we jointly use a CSLR and NMT system to perform translation on video inputs and evaluate end-to-end performance on the RWTH-PHOENIX-Weather 2014T dataset

6.2 Gloss2Text

We first perform translation on ground truth glosses taken from the RWTH-PHOENIX-Weather 2014T and ASLG-PC12 datasets. This allows us to assess the performance of our translation model when used with a perfect CSLR system, i.e. a CSLR system that produces gloss annotations with 100% accuracy. G2T is a text-to-text translation task that is novel and challenging compared to classic translation tasks between spoken languages because of the high linguistic variance between source and target sentences, scarcity of resources, and information loss or imprecision in the source sentence itself since it is merely an intermediate representation of the original phrase.

Experiments on RWTH-PHOENIX-Weather 2014T enable us to compare the performance of our Transformer on previous works using RNN-based models, and test our hypothesis that Transformers can enhance SLT. Experiments on ASLG-PC12 also allows us to explore the behavior of Transformers on a larger dataset. Moreover, Table 6.1 shows that the source and target corpora in ASLG-PC12 are more similar to each other with many shared vocabulary and a relatively high BLEU-4 score on raw data. This will also allow us to compare the performance of Transformers on a less challenging dataset.

The input phrases are tokenized to gloss level and we initialize the embedding matrix randomly, which is then trained in an end-to-end manner along with the whole model. For ASLG-PC12, many of the ASL glosses are English words with an added prefix so during data pre-processing we remove all such prefixes as we deem them unessential. We also set all words that appear less than 5 times during training as an unknown token which allows us to reduce the vocabulary size considerably, as shown in Table 6.1.

6.2.1 Model size

The original setup of the Transformer architecture in [63] uses 6 identical layers each in the encoder and the decoder to obtain their NMT results. However, our task may differ from

Table 6.1 Statistics of the ASLG-PC12 dataset before and after preprocessing.

	Raw data						Preprocessed data					
	Train		Dev		Test		Train		Dev		Test	
	ASL	en	ASL	en	ASL	en	ASL	en	ASL	en	ASL	en
Vocab.	15,782	21,600	4,323	5,634	2,150	2,609	5,906	7,712	1,163	1,254	394	379
Shared vocab.	10,048		2,652		1,296		4,941		899		287	
BLEU-4	20.97		21.16		20.63		38.87		38.74		38.37	

a standard machine translation task between two spoken languages so our first experiment trains Transformer models with 1, 2, 4 and 6 encoder-decoder (enc-dec) layers. All networks are trained with a batch size of 2,048 token and an initial learning rate of 1.

To choose the best model, we will mainly take into account the BLEU-4 score, as it is currently the most widely used metric in machine translation. Table 6.2 shows that on RWTH-PHOENIX-Weather 2014T, the architecture with 2 layers obtains the highest performance on both the development and testing sets. Moreover, a smaller model has the advantage of taking up less memory and computation time. Because our dataset is much smaller than those used in standard machine translation tasks, larger networks may be disadvantaged here. Repeating the same experiment on ASLG-PC12, we also find 2 layers to be the optimal model size. The ASLG-PC12 is a larger dataset, but the task at hand is simpler which may also explain why smaller networks are more suitable. We carry out the rest of our experiments using 2 enc-dec layers.

6.2.2 Batch size

Batch size refers to the number of training examples used per iteration. The recommended batch size by [31] to train Transformers is 4,096 tokens, which does not fit into our GPU memory. While [9] reports the best results with small batch size when training RNN-based

Table 6.2 G2T performance of Transformers on RWTH-PHOENIX-Weather 2014T with different number of enc-dec layers.

	Dev Set						Test Set					
Layers	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR
1	43.39	32.47	24.27	20.26	44.66	42.64	43.26	32.23	25.59	21.31	45.28	42.56
2	**45.31**	33.65	**26.73**	**22.23**	45.74	**43.92**	**44.57**	**33.08**	**26.14**	**21.65**	40.47	**42.97**
4	44.32	**32.87**	26.15	21.78	**45.86**	43.31	44.10	32.82	25.99	21.57	**45.44**	42.92
6	44.04	32.46	25.67	21.34	44.09	42.32	43.74	32.44	25.67	21.32	41.69	42.58

models for SLT, we train Transformer models using batch size 2,048, 1,024, 256, 128, 32 and 1 as Transformers might report different behavior.

We can also perform gradient accumulation which is approximately equivalent to increasing the batch size by the number of times we accumulate gradients without needing additional GPU memory. We perform additional experiments with batch size 2,048 and 2, 3, 5 or 10 gradient accumulations.

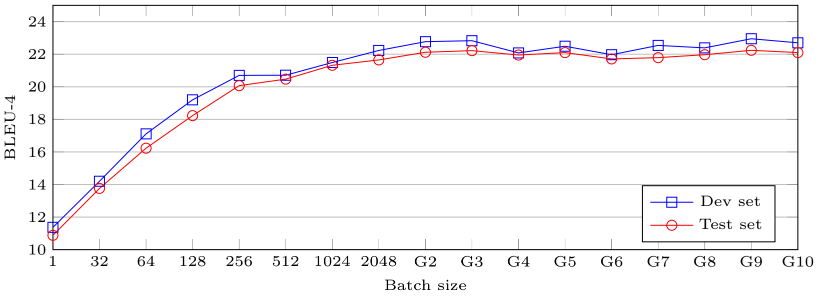

Figure 6.1 G2T performance on RWTH-PHOENIX-Weather 2014T with different batch size and gradient accumulation. Gn stands for batch size 2048 with n gradient accumulations.

Figure 6.1 confirms that the higher the batch size, the better. This is because in general, Transformers empirically train better on large batch sizes [48]. Batch size 2,048 gives the best performance when used with 3 gradient accumulations for both RWTH-PHOENIX-Weather 2014T and ASLG-PC12. We also invite others to try using an even bigger batch size on a larger GPU as using a bigger batch size may be more efficient than its approximate equivalent using gradient accumulation. The remaining experiments are performed with batch size 2,048 and 3 gradient accumulations.

6.2.3 Embedding schemes

[50] shows that tying the input and output embeddings during the training of language models may provide better performance. In our model, the decoder is also a language model that is conditioned on the encoding of the source sentence and the previous words of the generated sentence. We therefore tie the embeddings in the decoder by using a shared weight matrix for the input and output word embeddings.

In addition, pre-trained embeddings are a widely used method in NLP transfer learning where we initialize our models using pre-trained word embeddings. These word embeddings are typically trained in an unsupervised manner on a large corpus of text in the desired language, such as Wikipedia articles. They help bring in outside information at the start of training and can be useful when dealing with a small dataset.

Since German is a high-resource language where many German data resources exist, several pre-trained German embeddings are available publicly. We perform experiments on RWTH-PHOENIX-Weather 2014T using two popular word embeddings: GloVe[1] [47] and fastText [6]. GloVe handles words while fastText handles subwords and therefore provides better embeddings of rare words. To the best of our knowledge, pre-trained embeddings have never been used in SLT in previous works.

As shown in Table 6.3, there is only one matching token between the German glosses and the pre-trained embeddings, while over 90% of the words in the German text appear in both pre-trained embeddings. For this reason, in this part of the experiment we initialize with the pre-trained embeddings for the decoder only, and keep random initialization for the encoder. We do not freeze the embedding layers and fine-tune them during training for our task.

[1] We use pre-trained German vectors from https://deepset.ai/german-word-embeddings

Table 6.3 German and English pre-trained embeddings statistics

	GloVe (de)	fastText (de)	GloVe (en)	fastText (en)
Dimension	300	300	300	300
Source match	0.08%	0.08%	96.23%	94.64%
Target match	90.53%	94.57%	97.71%	96.32%

Table 6.4 shows that the new embedding schemes do not actually help in improving performance on RWTH-PHOENIX-Weather 2014T. It may be because pre-trained embeddings are shown to be more effective when used on the encoding layer [52], but we have no available pre-trained embeddings in German sign language glosses. Another possible reason is the difference between the domain of our dataset and of the corpus the embeddings were trained on, as our dataset has a specific domain of weather forecasts. We therefore keep random initialization of word embeddings for the rest of our experiments on RWTH-PHOENIX-Weather 2014T.

Table 6.4 G2T performance comparison using different embedding schemes on RWTH-PHOENIX-Weather 2014T.

	Dev Set						Test Set					
WE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR
Vanilla embedding	45.81	34.06	**27.05**	**22.49**	**46.68**	**44.35**	**45.29**	**33.74**	**26.70**	**22.22**	**46.08**	43.75
Tied decoder	**45.90**	**34.10**	26.98	22.31	46.76	44.51	45.05	33.38	26.31	21.74	45.83	43.45
GloVe	44.37	32.65	26.00	21.41	45.03	42.38	44.69	32.93	25.73	21.04	42.70	**44.61**
fastText	44.91	33.23	26.60	22.04	46.17	43.70	44.21	32.90	25.94	21.64	45.55	42.95

On ASLG-PC12, we also try tying the decoder embeddings as well as using English pre-trained GloVe and fastText embeddings. Table 6.3 shows that both GloVe and fastText English vectors have a reasonable overlap with the vocabulary of ASL glosses as well as the English targets. We therefore load pre-trained embeddings on only the decoder side as well as on both the encoder and decoder sides in our experiments.

Table 6.5 shows that using fastText pre-trained embeddings on the decoder improves performance compared to the vanilla embedding scheme, while using tied decoder embeddings without pre-trained embeddings gives the best performance in our experiment. Weight tying is more suited on this dataset likely because it can act as regularization and combat overfitting, while the previous dataset is more complex and therefore less prone to overfitting. We also performed an additional experiment using fastText pre-trained embeddings and weight tying on the decoder, but it does not surpass tied embeddings without pre-trained embeddings. For the remaining experiments on ASLG-PC12, we use tied decoder embeddings with random initialization.

Table 6.5 G2T performance comparison using different embedding schemes on ASLG-PC12.

	Dev Set						Test Set					
Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR
Vanilla embedding	90.15	84.92	80.27	75.94	94.72	94.58	90.49	85.64	81.31	77.33	94.75	95.16
Tied dec	**91.00**	**86.26**	**82.00**	**78.02**	**95.24**	**95.12**	**91.25**	**86.76**	**82.76**	**79.02**	**95.32**	**95.75**
GloVe dec	90.13	85.14	80.67	76.49	94.16	94.69	90.51	85.83	81.65	77.74	94.80	95.27
GloVe enc-dec	89.65	84.33	79.72	75.48	93.02	93.62	90.01	85.15	80.88	76.95	93.00	94.14
fastText dec	90.64	85.63	81.14	76.94	94.75	95.02	91.20	86.62	82.53	78.72	94.73	95.57
fastText enc-dec	90.02	85.01	80.56	76.41	93.68	94.10	90.94	86.58	82.01	76.23	93.61	94.42
fastText tied dec	90.16	85.26	80.85	76.72	95.03	94.60	90.44	85.25	81.69	77.28	95.11	95.04

6.2.4 Experiments on learning rate and warmup steps

A learning rate that is too low results in a notably slower convergence, but setting the learning rate too high risks leading the model to diverge. To prevent the model from diverging, we apply the Noam learning rate schedule where the learning rate increases linearly during the first training steps, or the warmup stage, then decreases proportionally to the inverse square root of the step number. The number of warmup steps is a hyperparameter that has shown to influence Transformer performance [48] therefore we run a hyperparameter search over the number of warmup steps and learning rate. We find the optimal combination to be an initial learning rate 0.5 with 3,000 warm-up steps for RWTH-PHOENIX-Weather 2014T and an initial learning rate of 0.2 and 8,000 warm-up steps for ASLG-PC12, which we use in the remaining experiments.

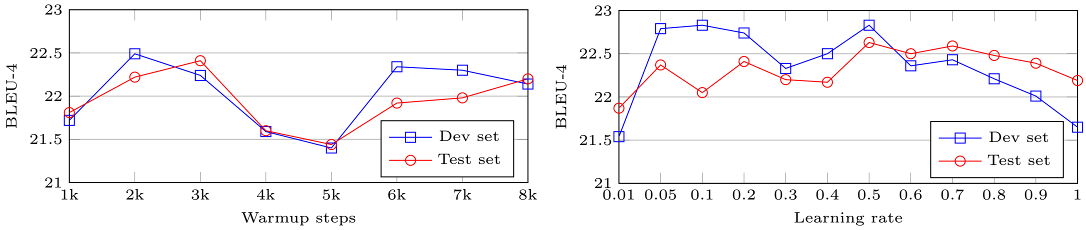

Figure 6.2 G2T performance on RWTH-PHEONIX-WEATHER 2014T with different warmup steps (left) and learning rate (right).

6.2.5 Beam width

A naive method for decoding a sequence of words is greedy search, where the model simply chooses the word with the highest probability at each time step of the sequence. However, this simple approach may be suitable for one time step, but becomes sub-optimal in the context of the entire sequence. Beam search is a widely used method to address this problem, in which at each time step, the decoder expands with all possible candidates and keeps a number of most likely sequences, or the beam width. Large beam widths do not always result in better performance and take more space in memory and decoding time. We therefore use our best performing model to decode using different beam widths and find the optimal beam width value to be 4 on RWTH-PHOENIX-Weather 2014T and 5 on ASLG-PC12.

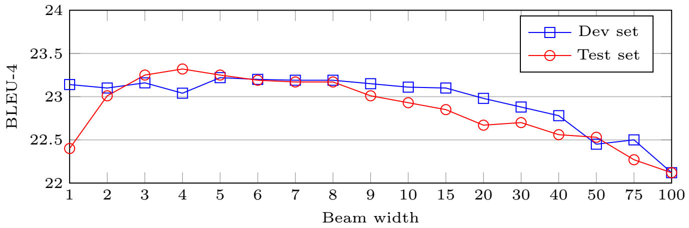

Figure 6.3 G2T decoding on RWTH-PHEONIX-WEATHER 2014T using different beam width. Beam width = 1 is equivalent to greedy search.

6.2.6 Ensemble decoding

Ensemble methods combine the predictions of multiple models and has shown to improve the overall performance in a wide range of tasks. We propose to employ ensemble decoding, where we use a group of models that have been trained separately during the decoding phase together. Ensemble decoding combines the output of different models by averaging their prediction distributions. We chose 9 models from our experiments that gave the highest BLEU-4 during testing on RWTH-PHOENIX-Weather 2014T. The number of models is chosen empirically, as using fewer models leads to less ensembling but including too many weaker models may decrease the quality of the ensemble model. These models are of the same architecture, but are initialized with different seeds and were trained using different batch sizes and/or learning rates. The models included in the ensemble give a BLEU-4 on testing between 22.92 and 23.41 individually.

Table 6.6 gives a performance comparison on RWTH-PHOENIX-Weather 2014T of the recurrent seq2seq model from [9], our single best performing model, and our ensemble model. We also provide the scores on the gloss annotations themselves to give an idea of the difficulty of this task.

Table 6.6 G2T on RWTH-PHEONIX-WEATHER 2014T final results. The results using an RNN-based seq2seq model is the one reported by [9].

	Dev Set						Test Set					
Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR
Raw data	13.01	6.23	3.03	1.71	24.23	13.69	11.88	5.05	2.41	1.36	22.81	12.12
RNN Seq2seq	44.40	31.93	24.61	20.16	46.02	–	44.13	31.47	23.89	19.26	45.45	–
Transformer	49.05	36.20	28.53	23.52	47.36	46.09	47.69	35.52	28.17	23.32	46.58	44.85
Transformer Ens.	**48.85**	**36.62**	**29.23**	**24.38**	**49.01**	**46.96**	**48.40**	**36.90**	**29.70**	**24.90**	**48.51**	**46.24**

Without any additional training, ensembling improves the BLEU-4 score on testing by over 1 point. Also, we report an improvement of over 5 BLEU-4 points on the current state-of-the-art. Moreover, a single Transformer gives an improvement of over 4 BLEU-4 points more than the state-of-the-art, which shows the advantage of Transformers over previous seq2seq networks for SLT.

We also use 5 of the best models from our experiments on ASLG-PC12 in an ensemble. These models report a BLEU-4 testing score between 81.72 and 82.41 individually. Table 6.7 compares the performance of our best single Transformer and ensemble model to the recurrent seq2seq model from [1]. The performance of a single Transformer surpasses the previous model by over 16 BLEU-4 points and the ensemble model reports an improvement of 0.46 BLEU-4 points over the single model. There is relatively less increase using an ensemble compared to a single model on this dataset, possibly because the score is already very high so there is less room for improvement without making substantial changes to the architecture, or there is less variance across different models so model behavior changes less in an ensemble.

Table 6.7 G2T on ASLG-PC12 final results. The results using an RNN-based seq2seq model is the one reported by [1].

	Dev Set						Test Set					
Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR
Raw data	54.60	39.67	28.92	21.16	76.11	61.25	54.19	39.26	28.44	20.63	75.59	61.65
Preprocessed data	69.25	56.83	46.94	38.74	83.80	78.75	68.82	56.36	46.53	38.37	83.28	79.06
RNN Seq2seq [1]	–	–	–	–	–	–	86.7	79.5	73.2	65.9	–	–
Transformer	**92.98**	**89.09**	83.55	**85.63**	82.41	95.93	92.98	**89.09**	85.63	82.41	95.87	**96.46**
Transformer Ens.	92.67	88.72	**85.22**	81.93	**96.18**	**95.95**	**92.88**	89.22	**85.95**	**82.87**	**96.22**	96.60

6.3 German Sign2Gloss2Text

We would also like to simulate an end-to-end system where both the tokenization into glosses and the translation of glosses to text are carried out by automatic methods. This is closer to what can be used in practice, since the system translates directly from sign language videos to spoken language. To achieve this, we use a STMC network to produce gloss annotations from sign language videos. We report the translation model's performance on the output of the STMC model on the development and the testing videos.

6.3.1 S2G → G2T

To begin, we use the best performing model from the last experiment on German G2T, and we simply feed the output of the STMC network to this model to obtain German translations. In Table 6.8 we can see that despite having no additional training, this model already obtains a relatively high score that beats the current state-of-the-art in German S2G2T by over 5 BLEU-4 points. The performance is slightly inferior to the prediction of ground truth glosses. There must be variance between the ground truth glosses and the STMC network outputs, which is why this model is better suited for G2T, but the two glosses have enough similarity for the model to provide good results despite never having seen STMC glosses during training.

6.3.2 Recurrent sequence-to-sequence networks

Since there is no existing work performing SLT with the STMC network, we also train attention-based seq2seq networks on glosses predicted by the STMC model for comparison. The seq2seq networks are built using four stacked layers of Gated Recurrent Units (GRU) [11], and we compare models that use Luong [39] and Bahdanau [4] attention mechanisms.

Table 6.8 shows that the recurrent seq2seq model obtains slightly better performance with Luong attention. What is surprising, however, is that for recurrent seq2seq models of similar architecture, our model that translates the output of the STMC network provides better results than the model in [9] that translates ground truth glosses. We will discuss this result further in the next subsection.

6.3.3 Transformer

Finally, we train Transformer models with the same architecture as the last experiment. We run a hyperparameter search over the learning rate and the beam size. We find an initial learning rate of 1 with 3,000 warm-up steps and beam size 4 to be the optimal one for this

Table 6.8 S2G2T performance. The first set of rows correspond to the current state-of-the-arts included for comparison.

	Dev Set						Test Set					
Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	METEOR
G2T [9]	44.40	31.93	24.61	20.16	46.02	–	44.13	31.47	23.89	19.26	45.45	–
S2G → G2T [9]	41.08	29.10	22.16	17.86	43.76	–	41.54	29.52	22.24	17.79	43.45	–
S2G2T [9]	42.88	30.30	23.03	18.40	44.14	–	43.29	30.39	22.82	18.13	43.80	–
S2G → G2T	46.75	34.99	27.79	23.06	47.29	45.23	47.49	35.89	28.62	23.77	47.32	45.54
Bahdanau	45.89	32.24	24.93	20.52	44.46	43.48	47.53	33.82	26.07	21.54	45.50	44.87
Luong	45.61	32.54	26.33	21.00	46.19	44.93	47.08	33.93	26.31	21.75	45.66	44.84
Transformer	48.27	35.20	27.47	22.47	46.31	44.95	48.73	36.53	29.03	24.00	46.77	45.78
Transformer Ens.	**50.31**	**37.60**	**29.81**	**24.68**	**48.70**	**47.45**	**50.63**	**38.36**	**30.58**	**25.40**	**48.78**	**47.60**

task. Then, we use our 8 best models in an ensemble as before to obtain the final result. Individually, the models in the ensemble give a BLEU-4 score between 23.51 and 24.00.

The end-to-end system of an STMC network with an ensemble of Transformers gives an improvement of over 7 BLEU-4 points on the current state-of-the-art for S2G2T SLT. A single improves performance of S2G2T using STMC by over 2 BLEU-4 points compared to recurrent seq2seq models. Appendix B provides translation examples that compare qualitatively the outputs of German G2T and S2G2T using Transformers.

Again, we observe that Transformers obtain better SLT performance by translating the STMC network output than ground truth glosses. This result is not restricted to Transformers, as we have seen previously recurrent seq2seq models report the same behavior. While the STMC network performs imperfect CSLR, its gloss predictions are better suited for SLT than ground-truth annotations and are more readily analyzed by the Transformer model. The glosses merely represent a simplified intermediate representation of the original sign language, so it is expected that even the ground truth glosses contain imprecision that negatively affect SLT performance. This result also gives a counter-example to the claim in [9] where they state G2T performance acts as an upper bound for end-to-end SLT. This novel result provides new directions for future research which we outline in the next chapter.

Chapter Seven

Conclusions

> *Translation is not a matter of words only: it is a matter of making intelligible a whole culture.*
>
> – Anthony Burgess, English writer and composer

7.1 Achievements

This thesis started by exploring the area of machine translation for sign language. We addressed several existing challenges and our examination of current works in SLT reveals that while latest research efforts have reported improvements in the tokenization of sign language videos, there has been no study to this date on improving translation of sign language glosses. The goal of this thesis was therefore to enhance the translation system in SLT. The recent success of Transformers on various NMT tasks inspired us to center our study around this architecture.

Our work is not the first to use Transformers in SLT [32, 43], however it is the first to successfully do so. Previous works focus on CSLR instead, which may explain why their Transformer models give weaker performance than RNN-based models. Transformers have a complex architecture and are especially sensitive to hyperparameter and model settings which make them harder to train, especially on a novel application.

The scientific achievements and contributions of this thesis are summarized below.

- We carried out the first thorough study and successful application of Transformers where we demonstrated how it outperforms previous NMT architectures for SLT
- We performed the first experiments using weight tying, transfer learning with spoken language data and ensemble learning in SLT
- We reported a series of baseline results of SLT with Transformers in various setups to underpin future research
- We applied a STMC model to SLT for the first time and demonstrated how end-to-end translation between videos and text can surpass translation from ground truth glosses
- We improved on the state-of-the-art results in German SLT on the RWTH-PHOENIX-Weather 2014T dataset for both sign language gloss to spoken language text translation and end-to-end sign language video to spoken language text translation, as well as in American SLT on the ASLG-PC12 dataset
- Research presented in this thesis produced a paper[1] submitted to peer-reviewed conferences

7.2 Future Work

We notably obtained superior performance in an end-to-end system using an STMC network to extract glosses from videos compared to a system that simulates perfect CSLR using ground truth glosses. As future work in S2G2T, we suggest jointly training the CSLR and NMT systems at the same time so that the CSLR model output glosses easily usable by the NMT model. We also suggest designing a gloss annotation scheme that optimizes translation to train the CSLR model on ground truth glosses adapted to make the NMT task simpler.

[1] https://arxiv.org/[redacted] Note for UA: the content of this thesis equally exists as a paper of 8 pages.

Glossary

Here is a list of specialist terms commonly used in this thesis.

D/deaf Different terms are generally accepted to describe people with hearing loss. 'Deaf' with uppercase D refers to people who identify as culturally Deaf rather than viewing deafness as an impairment, take pride in their Deaf identity, are actively engaged with the Deaf community, and often use sign language as their first language. 'deaf' with lowercase d describes people who refer to hearing loss as a medical condition, often do not have a strong involvement with the Deaf community, and may or may not use sign language as their preferred language.

Dropout is a regularization method where a randomly selected group of neurons in the neural network are ignored during a forward or backward pass.

Early stopping is a form of regularization where training is halted when performance on the development set degrades.

Epoch An epoch is one complete pass of the entire training dataset.

Embedding is a learned representation of some entity. In our case, word embeddings are a set of vectors that represent words in the model.

Gradient accumulation runs k smaller batches of size N before doing a backwards pass. This results in an equivalent batch size of $k \times N$.

Ground truth refers to data obtained by direct observation or empirically, as opposed to data obtained by inference.

Hyperparameter is a variable external to the model that are often used to help estimate model parameters. They can be set by the practitioner, or by using heuristics, and are turned to the given problem.

Learning rate controls how much the weights are updated at each step during training.

Neural network A typical feed-forward neural network (FNN) consists of several artificial neurons organized into layers. The input layer receives outside data the network will process. The output layer signals how the network responds to the input data. Between the input and output layers are one or more hidden layers used to process the data. The layers in the network are connected through weighted sums: a neuron's input is the weighted sum of all neurons in the previous layer, where the weights are learned by the network during training through backpropagation. Figure 7.1 gives the topology of a FNN as well as a recurrent neural network (RNN). Unlike FNNs, RNNs uses previous outputs of a layer as inputs again, which allow them to store internal memory and are useful for sequence learning tasks.

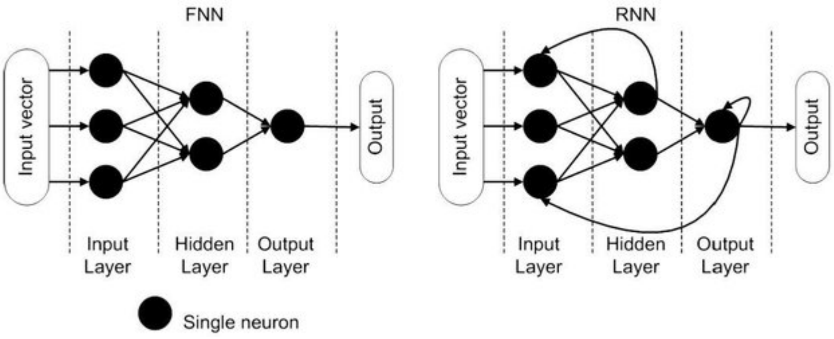

Figure 7.1 Traditional FNN and RNN. Figure from [35].

Parameter is a variable internal to the model that can be estimated from the given data. Example of model parameters include the weights in a neural network.

Regularization One issue that may arise in machine learning is overfitting, where the model fits the training data too closely and therefore does not generalize well on unseen data. Regularization refers to techniques that combat overfitting.

Transfer learning is a method to improve model performance where a model developed for a task is reused as the starting point for a model on a similar task.

Tokenization In classic NLP, tokenization is a preprocessing step to divide text into meaningful pieces, i.e words, subwords, sentences. In this thesis we refer to tokenization as the step "dividing" videos into meaningful glosses, which poses additional challenges as tokenization cannot be performed heuristically and the output of tokenization is often imprecise.

Word error rate (WER) is defined by the equation below, and is a measure often used in sign language recognition tasks among others. The lower the WER, the closer the outputs are to the ground truth.

$$\text{WER} = \frac{|\text{substitutions}| + |\text{deletions}| + |\text{insertions}|}{|\text{words in ground truth}|}$$

References

[1] Nikolaos Arvanitis, Constantinos Constantinopoulos, and Dimitrios Kosmopoulos. “Translation of Sign Language Glosses to Text Using Sequence-to-Sequence Attention Models”. 2019.

[2] Stylianos Asteriadis, George Caridakis, and Kostas Karpouzis. “Non-manual cues in automatic sign language recognition”. *Personal and Ubiquitous Computing* 18 (2012). DOI: 10.1007/s00779-012-0615-1.

[3] Vassilis Athitsos, Carol Neidle, Stan Sclaroff, Joan P. Nash, Alexandra Stefan, Ashwin Thangali, Haijing Wang, and Quan Yuan. “Large Lexicon Project : American Sign Language Video Corpus and Sign Language Indexing / Retrieval Algorithms”. 2010.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. “Neural Machine Translation by Jointly Learning to Align and Translate”. *CoRR* abs/1409.0473 (2014).

[5] Satanjeev Banerjee and Alon Lavie. “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”. *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization.* Ann Arbor, Michigan: Association for Computational Linguistics, 2005, pp. 65–72. URL: https://www.aclweb.org/anthology/W05-0909.

[6] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. “Enriching Word Vectors with Subword Information”. *arXiv preprint arXiv:1607.04606* (2016).

[7] Patrick Buehler, Andrew Zisserman, and Mark Everingham. “Learning sign language by watching TV (using weakly aligned subtitles)”. *2009 IEEE Conference on Computer Vision and Pattern Recognition* (2009), pp. 2961–2968.

[8] Necati Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. “SubUNets: End-to-End Hand Shape and Continuous Sign Language Recognition”. 2017. DOI: 10.1109/ICCV.2017.332.

[9] Necati Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. “Neural Sign Language Translation”. 2018. DOI: 10.1109/CVPR.2018.00812.

[10] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches". *ArXiv* abs/1409.1259 (2014).

[11] Junyoung Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling". *ArXiv* abs/1412.3555 (2014).

[12] Helen Cooper, Eng-Jon Ong, Nicolas Pugeault, and Richard Bowden. "Sign Language Recognition using Sub-Units". *Journal of Machine Learning Research* 13 (July 2012), pp. 2205–2231.

[13] Andy Cornes and Jemina Napier. "Challenges of mental health interpreting when working with deaf patients". *Australasian Psychiatry* 13.4 (2005). PMID: 16403140, pp. 403–407. DOI: 10.1080/j.1440-1665.2005.02218.x. eprint: https://www.tandfonline.com/doi/pdf/10.1080/j.1440-1665.2005.02218.x. URL: https://www.tandfonline.com/doi/abs/10.1080/j.1440-1665.2005.02218.x.

[14] Onno Crasborn, Richard Bank, Inge Zwitserlood, Els van der kooij, Anne Meijer, Anna Sáfár, and Ellen Ormel. *Annotation Conventions for the Corpus NGT, version 3*. Feb. 2015. DOI: 10.13140/RG.2.1.1779.4649.

[15] Onno Crasborn, Johanna Mesch, Dafydd Waters, A. Nonhebel, Els van der Kooij, Bencie Woll, and Brita Bergman. "Sharing sign language data online: Experiences from the ECHO project". 2007.

[16] Runpeng Cui, Hu Liu, and Changshui Zhang. "Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization". *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 1610–1618.

[17] Eleni Efthimiou, Stavroula-Evita Fotinea, Thomas Hanke, John R. W. Glauert, Richard Bowden, Annelies Braffort, Christophe Collet, Petros Maragos, and François Lefebvre-Albaret. "Sign Language technologies and resources of the Dicta-Sign project". 2012.

[18] Jens Forster, Oscar Koller, Christian Oberdörfer, Yannick L. Gweth, and Hermann Ney. "Improving Continuous Sign Language Recognition: Speech Recognition Techniques and System Design". *SLPAT*. 2013.

[19] Jens Forster, Christian Oberdörfer, Oscar Koller, and Hermann Ney. "Modality Combination Techniques for Continuous Sign Language Recognition". June 2013. DOI: 10.1007/978-3-642-38628-2_10.

[20] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann Dauphin. "Convolutional Sequence to Sequence Learning". *ICML*. 2017.

[21] Dan Guo, Shengeng Tang, and Meng Wang. "Connectionist Temporal Modeling of Video and Language: a Joint Model for Translation and Sign Labeling". *IJCAI*. 2019.

[22] Felix Hieber and Tobias Domhan. "Train Neural Machine Translation Models with Sockeye" (2017). URL: https://aws.amazon.com/blogs/machine-learning/train-neural-machine-translation-models-with-sockeye/.

[23] Jie Huang, Wengang Zhou, Houqiang Li, and Weiping Li. "Sign Language Recognition using 3D convolutional neural networks". *2015 IEEE International Conference on Multimedia and Expo (ICME)* (2015), pp. 1–6.

[24] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li. "Video-based Sign Language Recognition without Temporal Segmentation". *AAAI*. 2018.

[25] Trevor Johnston. "Auslan Corpus Annotation Guidelines". 2013.

[26] Trevor Alexander Johnston. "Corpus linguistics and signed languages: no lemmata, no corpus". English. *Proceedings of the Sixth International Language Representation and Evaluation Conference*. Ed. by E. Efthimiou O. Crasborn and I. Zwitserlood. 2008, pp. 82–87.

[27] Lukasz Kaiser, Aidan N. Gomez, and François Chollet. "Depthwise Separable Convolutions for Neural Machine Translation". *ArXiv* abs/1706.03059 (2017).

[28] Nal Kalchbrenner and Phil Blunsom. "Recurrent Continuous Translation Models". *EMNLP*. 2013.

[29] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aäron van den Oord, Alex Graves, and Koray Kavukcuoglu. "Neural Machine Translation in Linear Time". *ArXiv* abs/1610.10099 (2016).

[30] Diederik Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". *International Conference on Learning Representations* (2014).

[31] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. "OpenNMT: Open-Source Toolkit for Neural Machine Translation". *Proc. ACL*. 2017. DOI: 10.18653/v1/P17-4012. URL: https://doi.org/10.18653/v1/P17-4012.

[32] Sangki Ko, Chang Kim, Hyedong Jung, and Choongsang Cho. "Neural Sign Language Translation Based on Human Keypoint Estimation". *Applied Sciences* 9 (2019), p. 2683. DOI: 10.3390/app9132683.

[33] Oscar Koller, Jens Forster, and Hermann Ney. "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers". *Computer Vision and Image Understanding* 141 (2015), pp. 108–125. DOI: 10.1016/j.cviu.2015.09.013.

[34] Oscar Koller, Sepehr Zargaran, and Hermann Ney. "Re-Sign: Re-Aligned End-to-End Sequence Modelling with Deep Recurrent CNN-HMMs". *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 3416–3424.

[35] Andrej Krenker, Janez Bester, and Andrej Kos. "Introduction to the Artificial Neural Networks". Apr. 2011. ISBN: 978-953-307-243-2. DOI: 10.5772/15751.

[36] Jolanta Lapiak. "ASL sentence for: It takes 17 muscles to smile and 43 to frown." *Handspeak* (2020). URL: https://www.handspeak.com/translate/index.php?id=288.

[37] Jeroen Lichtenauer, Emile Hendriks, and Marcel Reinders. "Sign Language Recognition by Combining Statistical DTW and Independent Classification". *IEEE transactions on pattern analysis and machine intelligence* 30 (Dec. 2008), pp. 2040–6. DOI: 10.1109/TPAMI.2008.123.

[38] Chin-Yew Lin. "ROUGE: A Package for Automatic Evaluation of summaries". 2004, p. 10.

[39] Minh-Thang Luong, Hieu Pham, and Christoper Manning. "Effective Approaches to Attention-based Neural Machine Translation" (Aug. 2015). DOI: 10.18653/v1/D15-1166.

[40] Pavlo Molchanov, Xiaodong Yang, Shalini Gupta, Kihwan Kim, Stephen Tyree, and Jan Kautz. "Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D Convolutional Neural Networks". June 2016, pp. 4207–4215. DOI: 10.1109/CVPR.2016.456.

[41] Sara Morrissey, Harold L. Somers, Robert G. Smith, Shane Innzna Gilchrist, and Sandipan Dandapat. "Building a sign language corpus for use in machine translation". *LREC 2010*. 2010.

[42] Karen Nakamura. "About American Sign Language". *Deaf Resource Library* (1995).

[43] Alptekin Orbay and Lale Akarun. "Neural Sign Language Translation by Learning Tokenization". *ArXiv* abs/2002.00479 (2020).

[44] Achraf Othman and Mohamed Jemni. "English-ASL Gloss Parallel Corpus 2012: ASLG-PC12". 2012.

[45] Kishore Papineni, Salim Roukos, Todd Ward, and Wei Jing Zhu. "BLEU: a Method for Automatic Evaluation of Machine Translation" (2002). DOI: 10.3115/1073083.1073135.

[46] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–

8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[47] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. “GloVe: Global Vectors for Word Representation”. *Empirical Methods in Natural Language Processing (EMNLP)*. 2014, pp. 1532–1543. URL: http://www.aclweb.org/anthology/D14-1162.

[48] Martin Popel and Ondřej Bojar. “Training Tips for the Transformer Model”. *The Prague Bulletin of Mathematical Linguistics* 110 (2018). DOI: 10.2478/pralin-2018-0002.

[49] Janet L. Pray and I. King Jordan. “The Deaf Community and Culture at a Crossroads: Issues and Challenges”. *Journal of Social Work in Disability & Rehabilitation* 9.2-3 (2010). PMID: 20730674, pp. 168–193. DOI: 10.1080/1536710X.2010.493486. eprint: https://doi.org/10.1080/1536710X.2010.493486. URL: https://doi.org/10.1080/1536710X.2010.493486.

[50] Ofir Press and Lior Wolf. “Using the Output Embedding to Improve Language Models”. *ArXiv* abs/1608.05859 (2016).

[51] Siegmund Prillwitz. “HamNoSys. Version 2.0; Hamburg Notation System for Sign Language. An Introductionary Guide”. 1989.

[52] Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. “When and Why Are Pre-Trained Word Embeddings Useful for Neural Machine Translation?” *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 529–535. DOI: 10.18653/v1/N18-2084. URL: https://www.aclweb.org/anthology/N18-2084.

[53] Rung-Huei Liang and Ming Ouhyoung. “A real-time continuous gesture recognition system for sign language”. *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*. Apr. 1998, pp. 558–567. DOI: 10.1109/AFGR.1998.671007.

[54] Adam Schembri and Onno Crasborn. “Issues in creating annotation standards for sign language description”. Jan. 2010, ADD.

[55] Christoph Schmidt, Oscar Koller, Hermann Ney, Thomas Hoyoux, and Justus Piater. “Using viseme recognition to improve a sign language translation system”. 2013.

[56] Maria Fernanda Neves Silveira de Souza, Amanda Miranda Brito Araújo, Luiza Fernandes Fonseca Sandes, Daniel Antunes Freitas, Wellington Danilo Soares, Raquel Schwenck de Mello Vianna, and Arlen Almeida Duarte de Sousa. “Main difficulties and obstacles faced by the deaf community in health access: an integrative literature re-

view". pt. *Revista CEFAC* 19 (2017), pp. 395–405. ISSN: 1516-1846. URL: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1516-18462017000300395&nrm=iso.

[57] Thad Starner and Massachusetts Group. "Visual Recognition of American Sign Language Using Hidden Markov Models" (May 1995).

[58] Thad Starner, Joshua Weaver, and Alex Pentland. "Real-time American sign language recognition using desk and wearable computer based video". *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 20 (Jan. 1999), pp. 1371–1375. DOI: 10.1109/34.735811.

[59] Daniel Stein, Christoph Schmidt, and Hermann Ney. "Analysis, preparation, and optimization of statistical sign language machine translation". *Machine Translation* 26 (2012), pp. 325–357.

[60] William C. Stokoe. "Sign language structure: an outline of the visual communication systems of the American deaf." *Journal of deaf studies and deaf education* 10 1 (1960), pp. 3–37.

[61] Ilya Sutskever, Oriol Vinyals, and Quoc Le. "Sequence to Sequence Learning with Neural Networks". *Advances in Neural Information Processing Systems* 4 (Sept. 2014).

[62] Alaa Tharwat, Tarek Gaber, MK Shahin, Basma Refaat, and Aboul Ella Hassanien Ali. "SIFT-based Arabic Sign Language Recognition System". *The 1st Afro-European Conference for Industrial Advancement,* Addis Ababa, Ethiopia, Nov. 2014.

[63] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. "Attention is All you Need". *NIPS*. 2017.

[64] Amy T. Wilson and Rowena E. Winiarczyk. "Mixed Methods Research Strategies With Deaf People: Linguistic and Cultural Challenges Addressed". *Journal of Mixed Methods Research* 8.3 (2014), pp. 266–277. DOI: 10.1177/1558689814527943. eprint: https://doi.org/10.1177/1558689814527943. URL: https://doi.org/10.1177/1558689814527943.

[65] Q X Yang. "Chinese sign language recognition based on video sequence appearance modeling". *2010 5th IEEE Conference on Industrial Electronics and Applications* (2010), pp. 1537–1542.

[66] Farhad Yasir, P. W. Chandana Prasad, Abeer Alsadoon, and Amr Elchouemi. "SIFT based approach on Bangla sign language recognition". *2015 IEEE 8th International Workshop on Computational Intelligence and Applications (IWCIA)* (2015), pp. 35–39.

[67] Zahoor Zafrulla, Helene Brashear, Thad Starner, Harley Hamilton, and Peter Presti. "American sign language recognition with the kinect". Nov. 2011, pp. 279–286. DOI: 10.1145/2070481.2070532.

[68] Jihai Zhang, Wengang Zhou, and Houqiang Li. “A Threshold-based HMM-DTW Approach for Continuous Sign Language Recognition”. *ACM International Conference Proceeding Series* (July 2014). DOI: 10.1145/2632856.2632931.

[69] Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. “Spatial-Temporal Multi-Cue Network for Continuous Sign Language Recognition”. *arXiv e-prints*, arXiv:2002.03187 (2020), arXiv:2002.03187. arXiv: 2002.03187 `[cs.CV]`.

APPENDICES

Appendix A

Comparison of German G2T and S2G2T outputs

Because quantitative metrics provide only a limited evaluation of translation performance, manual evaluation by viewing the translation outputs directly may give a better assessment of the quality of translations. In Table A.1 we share examples of translation on the RWTH-PHEONIX-WEATHER 2014T dataset by the G2T and S2G2T networks accompanied by the respective gloss annotations, ground truth German translation, and English translations for the reader[1].

The examples show that the translations are of generally good quality, even for those with lower BLEU-4 scores. Most translations may have slight differences in word choice that do not change the overall meaning of the sentence, or present grammatical errors in German. As for the comparison between the G2T and S2G2T networks, there does not seem to be a clear pattern between cases where S2G2T outperforms G2T and vice versa. The difference between ground truth and predicted glosses are also often within a single gloss. Overall, qualitative results are quite satisfactory for human comprehension and are encouraging for practical applications of sign language translation.

[1] Best viewed electronically or printed on A3 paper

Table A.1 Qualitative comparison of G2T and S2G2T on RWTH-PHEONIX-WEATHER 2014T. GT refers to the ground truth German translation.

	Gloss	German	BLEU-4
GT:		und nun die wettervorhersage für morgen samstag den zweiten april .	
		(and now the weather forecast for tomorrow saturday the second of april .)	
G2T:	JETZT WETTER WIE-AUSSEHEN MORGEN SAMSTAG ZWEITE APRIL	und nun die wettervorhersage für morgen samstag den zweiten april .	100.00
	(NOW WEATHER LIKE-LOOKING TOMORROW SATURDAY SECOND APRIL)	(and now the weather forecast for tomorrow saturday the second of april .)	
S2G2T:	JETZT WETTER WIE-AUSSEHEN MORGEN SAMSTAG ZWEITE APRIL	und nun die wettervorhersage für morgen samstag den zweiten april .	100.00
	(NOW WEATHER LIKE-LOOKING TOMORROW SATURDAY SECOND APRIL)	(and now the weather forecast for tomorrow saturday the second of april .)	
GT:		ähnliches wetter auch am donnerstag .	
		(similar weather on thursday .)	
G2T:	GLEICH WETTER AUCH DONNERSTAG	ähnliches wetter auch am donnerstag .	100.00
	(SAME WEATHER ON THURSDAY)	(similar weather on thursday .)	
S2G2T:	GLEICH WETTER DONNERSTAG	ähnliches wetter dann auch am donnerstag .	48.89
	(SAME WEATHER THURSDAY)	(similar weather then on thursday .)	
GT:		der wind weht meist schwach aus unterschiedlichen richtungen .	
		(the wind usually blows weakly from different directions .)	
G2T:	WIND SCHWACH UNTERSCHIED KOMMEN	der wind weht meist nur schwach aus unterschiedlichen richtungen .	65.80
	(WIND WEAK DIFFERENCE COME)	(the wind usually blows only weakly from different directions .)	
S2G2T:	WIND SCHWACH UNTERSCHIED	der wind weht schwach aus unterschiedlichen richtungen .	61.02
	(WIND WEAK DIFFERENCE)	(the wind is blowing weakly from different directions .)	
GT:		am montag ist es wechselhaft mal sonne mal wolken vor allem in der osthälfte einige kräftige schauer und gewitter .	
		(on monday it is changeable sometimes sun sometimes clouds especially in the east half some strong showers and thunderstorms .)	
G2T:	MONTAG WECHSELHAFT MAL SONNE WOLKE BESONDERS REGION SCHAUER GEWITTER	am montag mal sonne mal wolken vor allem in der osthälfte einzelne schauer und gewitter .	55.59
	(MONDAY VARIABLE TIMES SUN CLOUD PARTICULARLY REGION SHOWER THUNDERSTORM)	(on monday sometimes sun sometimes clouds especially in the east half showers and thunderstorms)	
S2G2T:	MONTAG WECHSELHAFT MEHR SONNE WOLKE BESONDERS REGION SCHAUER GEWITTER	am montag mal sonne mal wolken vor allem in der nordhälfte schauer und gewitter .	49.38
	(MONDAY VARIABLE MORE SUN CLOUD PARTICULARLY REGION SHOWER THUNDERSTORM)	(on monday sometimes sun sometimes clouds especially in the north half showers and thunderstorms .)	
GT:		sonnig geht es auch ins wochenende samstag ein herrlicher tag mit temperaturen bis siebzehn grad hier im westen .	
		(the weekend is also sunny and saturday is a wonderful day with temperatures up to seventeen degrees here in the west .)	
G2T:	WOCHENENDE SONNE SAMSTAG SCHOEN TEMPERATUR BIS SIEBZEHN GRAD REGION	und am wochenende da scheint die sonne bei temperaturen bis siebzehn grad .	13.49
	(WEEKEND SUN SATURDAY NICE TEMPERATURE UNTIL SEVENTEEN DEGREE REGION)	(and on the weekend the sun shines at temperatures up to seventeen degrees .)	
S2G2T:	WOCHENENDE SONNE SAMSTAG TEMPERATUR BIS SIEBZEHN GRAD REGION	am wochenende scheint die sonne bei temperaturen bis siebzehn grad .	12.55
	(WEEKEND SUN SATURDAY TEMPERATURE UNTIL SEVENTEEN DEGREE REGION)	(on the weekend sun shines at temperatures up to seventeen degrees .)	
GT:		es gelten entsprechende warnungen des deutschen wetterdienstes .	
		(appropriate warnings from the german weather service apply .)	
G2T:	IX SCHON WARNUNG DEUTSCH WETTER DIENST STURM KOENNEN	es bestehen entsprechende unwetterwarnungen des deutschen wetterdienstes .	38.26
	(IX ALREADY WARNING GERMAN WEATHER SERVICE STORM CAN)	(severe weather warnings from the german weather service exist .)	
S2G2T:	DANN IX SCHON WARNUNG DEUTSCH WETTER STURM KOENNEN	es gelten entsprechende warnungen des deutschen wetterdienstes .	100.00
	(THEN IX ALREADY WARNING GERMAN WEATHER STORM CAN)	(appropriate warnings from the german weather service apply .)	
GT:		richtung osten ist es meist sonnig .	
		(it is mostly sunny towards the east .)	
G2T:	OST MEISTENS SONNE	im osten bleibt es meist sonnig .	43.47
	(MOST EAST SUN)	(in the east it mostly stays sunny .)	
S2G2T:	OST REGION MEISTENS SONNE	im osten ist es meist sonnig .	80.91
	(MOST REGION EAST SUN)	(in the east it is mostly sunny .)	
GT:		am sonntag im norden und an den alpen mal sonne mal wolken und ab und an schauer sonst ist es recht freundlich .	
		(on sunday in the north and in the alps sometimes sun sometimes clouds and occasionally showers otherwise it is quite pleasant .)	
G2T:	SONNTAG NORD ALPEN IX SONNE WOLKE SCHAUER SONST REGION FREUNDLICH	am sonntag im norden und an den alpen mal sonne mal wolken sonst ist es freundlich .	56.31
	(SUNDAY NORTH ALPS IX SUN CLOUD SHOWER OTHERWISE REGION FRIENDLY)	(on sunday in the north and in the alps sometimes sun sometimes clouds otherwise it is pleasant .)	
S2G2T:	SONNTAG NORD BERG IX SONNE WOLKE SCHAUER SONST REGION FREUNDLICH	am sonntag im norden und an den alpen mal sonne mal wolken und einzelne schauer sonst ist es freundlich .	67.30
	(SUNDAY NORTH MOUNTAIN IX SUN CLOUD SHOWER OTHERWISE REGION FRIENDLY)	(on sunday in the north and in the alps sometimes sun sometimes clouds and a few showers otherwise it is pleasant .)	
GT:		am tag elf grad im vogtland und einundzwanzig grad am oberrhein .	
		(during the day eleven degrees in vogtland and twenty one degrees in upper rhine .)	
G2T:	AM-TAG ELF VOGEL LAND	elf grad am oberrhein .	18.74
	(IN-THE-DAY ELEVEN BIRD LAND)	(eleven degrees in upper rhine .)	
S2G2T:	ELF VOGEL ZWANZIG	am tag elf grad im vogtland und zwanzig grad im vogtland .	54.91
	(ELEVEN BIRD TWENTY)	(during the day eleven degrees in vogtland and twenty degrees in vogtland .)	
GT:		es kann noch leicht schneien in den nächsten stunden in sachsen oder auch am alpenrand und in bayern .	
		(it can still snow lightly in the next few hours in saxony or on the border of the alps and in bavaria .)	
G2T:	BISSCHEN SCHNEE IN-KOMMEND SACHSEN ODER BERG REGION BAYERN REGION MOEGLICH SCHNEE	aber es fällt etwas schnee am alpenrand und in sachsen oder schneeregen .	18.23
	(BIT SNOW IN-COMING SAXONY OR MOUNTAIN REGION BAVARIA REGION POSSIBLE SNOW)	(but there is some snow on the border of the alps and in saxony or sleet .)	
S2G2T:	ABER BISSCHEN SCHNEE SACHSEN ODER BERG BAYERN MOEGLICH SCHNEE	aber es wird ein bisschen schneien am alpenrand und in sachsen oder schnee .	19.16
	(BUT BIT SNOW SAXONY OR MOUNTAIN BAVARIA POSSIBLE SNOW)	(but there will be a little snow on the border of the alps and in saxony or snow .)	

Appendix B

Qualitative G2T Results on ASLG-PC12

Table B.1 provides examples of SLT output on the ASLG-PC12 dataset. Here we can see how ASL glosses include prefixes that are not necessary to encapture the meaning of the phrase, which we have removed during data pre-processing before training. With a BLEU-4 testing score of 82.87, most predictions by our system are very close to the target English phrases and are able to convey the same meaning. We have also selected translation examples with lower BLEU-4 score and we can see that common errors include mistranslation of numbers and proper nouns. These are likely corner cases with infrequent examples during training.

Table B.1 Examples of ASL translation with varying BLEU-4 scores

		BLEU-4
ASL:	X-I BE DESC-PARTICULARLY DESC-GRATEFUL FOR EUROPEAN PARLIAMENT X-POSS DRIVE ROLE WHERE BALTIC SEA COOPERATION BE CONCERN	100.00
GT:	i am particularly grateful for the european parliament’s driving role where the baltic sea cooperation is concerned .	
Pred:	i am particularly grateful for the european parliament’s driving role where the baltic sea cooperation is concerned .	
ASL:	DESC-REFORE , DESC-MUCH WORK NEED TO BE DO IN ORDER TO DESC-FURR SIMPLIFY RULE	100.00
GT:	therefore , much work needs to be done in order to further simplify the rules .	
Pred:	therefore , much work needs to be done in order to further simplify the rules .	
ASL:	THIS PRESSURE BE DESC-PARTICULARLY DESC-GREAT ALONG UNION X-POSS DESC-SOURN AND DESC-EASTERN BORDER	100.00
GT:	this pressure is particularly great along the union’s southern and eastern borders .	
Pred:	this pressure is particularly great along the union’s southern and eastern borders .	
ASL:	MORE WOMAN DIE FROM AGGRESSION DESC-DIRECT AGAINST X-Y THAN DIE FROM CANCER .	73.15
GT:	more women die from the aggression directed against them than die from cancer .	
Pred:	more women die from aggression directed against them than die from cancer .	
ASL:	X-IT FUEL WAR IN CAMBODIUM IN 1990 AND X-IT BE ENEMY DEMOCRACY	25.89
GT:	it fuelled the war in cambodia in the 1990s and it is the enemy of democracy .	
Pred:	it fuel war in the cambodium in 1990 and it is an enemy of democracy .	
ASL:	DESC-N CHIEF INVESTIGATOR X-HIMSELF BE TARGET AND HOUSE CARD COLLAPSE .	21.29
GT:	then the chief investigator himself is targeted and the house of cards collapses .	
Pred:	then chief investigator himself is a target and a house card collapse .	
ASL:	U , X-WE TAKE DESC-DUE NOTE X-YOU OBSERVATION . AMENDMENT THANK X-YOU MR	15.93
GT:	otherwise we have to vote on the corresponding part of amendment thank you mrs ţicău , we take due note of your observation .	
Pred:	mr president , we took due note of your observation .	