NEURAL MACHINE TRANSLATION FOR BRITISH SIGN LANGUAGE

Dissertation

Chun Ting Justin LO
Author

Fernando Alva Manchego
Supervisor

Abstract

This dissertation explores the adaptation of existing neural machine translation technologies for British Sign Language (BSL), with the aim of improving accessibility and preserving linguistic identity. The study evaluates four sign-to-text translation approaches on BSL datasets, shedding light on their performance.

Additionally, it emphasizes the challenges posed by data scarcity in sign language research and the importance of appropriate data pre-processing methods. The research identifies key research gaps, which is the need for the development of more context-aware models training approach for scalable interpretation-based datasets, paving the way for future investigations.

In conclusion, this work contributes to the development of sign language translation technology, addressing the needs of the deaf and hard-of-hearing community and providing insights for future research directions.

Acknowledgment

I would personally like to thank my supervisor Mr. Fernando Alva Manchego and the School of Computer Science and Informatics at Cardiff University for their persistent aid in the completion of this project.

Table of Contents

1 Introduction

1.1 Motivation

Sign Languages (SLs) occupy a unique and indispensable role as the primary means of communication for the deaf and the hard of hearing(DHH) worldwide(Núñez-Marcos et al., 2023). These non-verbal speech communication languages serve as vital conduits for a substantial segment of the global population. According to the World Federation of the Deaf, there are an estimated 70 million deaf people and about 300 sign languages (SLs) in the world (UN, 2021).

Despite the significance and ubiquity of SLs, there exists a disconcerting trend wherein some Deaf individuals are increasingly encouraged to embrace alternative communication methods, such as lip-reading or text-based exchanges. This phenomenon is compounded by the exclusion of sign languages from modern spoken language technologies, which further marginalizes sign language usage in favor of spoken languages(K. Yin et al., 2021). The consequences are profound, extending beyond the realm of mere accessibility; they encroach upon questions of linguistic identity and cultural preservation.

Indeed, with the advancement in machine translation and speech translation tasks for spoken languages, there is increasing research effort extending these technologies to SLs(K. Yin et al., 2021). One of the machine translation tasks is text/speech to sign language (SL) translation, which is generally called Sign Language Translation (SLT), which is the focus of this dissertation. According to Núñez-Marcos et al.(2023), most research on this topic has worked on a small-scale German Sign Language (GSL) dataset, namely PHOENIX-2014T(Camgoz et al., 2018), for benchmarking SLT models.

Among SLs, British Sign Language (BSL) boasts 151,000 users, with 87,000 of them identifying as deaf, as documented by the British Deaf Association(2011). Nonetheless, there is limited work on BSL overall. As a student currently based in the United Kingdom, I am driven by the

desire to explore the feasibility of applying existing SL machine translation technologies to BSL.

1.2 Objectives

To explore the feasibility of applying existing SL neural machine translation technologies to BSL, this paper comprises two major objectives:

(i) To explore approaches to train BSL translation (sign-to-text) models

(ii) To evaluate the BSL datasets, data pre-processing methods, and training approaches

Specifically, the SLT model trained using four different approaches(Camgöz et al., 2020; Tarrés et al., 2023; Voskou et al., 2021; A. Yin et al., 2023) was investigated, and replicated on two BSL datasets.

1.3 Contributions

This dissertation makes several possible contributions to the field of neural machine translation for sign language and large-scale BSL datasets, which can be summarized as follows:

(i) **Replication and evaluation of Transformer training approaches**

Evaluation especially regarding the performance of four Transformer-based training approaches on BSL datasets is conducted, which provides brief insights into the current condition of the development of neural sign language translation model training approaches

(ii) **Dataset evaluation and Insights into data scarcity**

This dissertation provides a deeper understanding of the challenges posed by data scarcity in SL research. It emphasizes the importance of large domains of discourse and the potential of scalable interpretation-based datasets. These findings may contribute to the foundation for future work in data collection and dataset construction.

(iii) **Exploration of data pre-processing methods**

By investigating the significance of data pre-processing methods, this dissertation

highlights the importance of selecting appropriate feature extraction and tokenization methods.

(iv) **Identification of research gap**

This dissertation recognizes some of the main limitations of available data and training approaches, suggesting possible future research directions. Specifically, it acknowledges the need for further experimentation with multi-encoder Transformer approaches and the development of more context-aware models for scalable interpretation-based datasets.

In conclusion, this dissertation highlights areas where further research can make significant contributions to the field of sign language translation. The code for data pre-processing and re-arrangement is publicly available at https://github.com/justinlctstudy96/nmt_bsl.

2 Background

2.1 History of Sign Language

Throughout modern history, sign languages have been struggling to be recognized as natural, independent, and well-defined languages. Spoken languages were so dominant that sign languages were deemed as a kind of development hindering speech skills (Moryossef & Goldberg, 2021). In 1880, the Second International Congress on Education of the Deaf banned teaching signed languages to favor speech therapy instead. Sign languages were not recognized until 1960, when sign languages started gaining recognition due to the seminal work on American Sign Language (ASL) (Landar, 1961), resulting in a new research area.

Nonetheless, there are currently still antiquated notions that deprioritize sign languages causing neglect of the linguistic needs of their users (Humphries et al., 2016). Studies show that a lot of DHH children still grow up with spoken languages only (Murray et al., 2020). They cannot get enough access to a first language during their critical language acquisition period, adversely affecting their cognitive, linguistic, socio-emotional, and academic development(Hall et al., 2017).

There is a broad barrier between signers and non-signers. In many situations, DHH people are ignored or forced to use alternative communication tools instead of sign language. Many alternatives are not comfortable for the DHH people, such as writing down the message in the form of spoken language or using some special gloves for signs (Núñez-Marcos et al., 2023).

2.2 Characteristics of Sign Language

Sign Languages are not just gesture systems, they have their own grammar and vocabulary (Landar, 1961). It is common for each country to have its own sign language with its unique

vocabulary, possibly sharing some similarities such as grammar (Stokoe, 1980). Even for countries sharing the same spoken language, the sign language system can be different. For instance, in the case of English-speaking countries, there are British Sign Language (BSL), American Sign Language (ASL), and Australian Sign Language (Auslan).

Sign Languages are expressed through articulators, which means information is conveyed using parts of the body. Articulators can be classified into manual and non-manual (Núñez-Marcos et al., 2023). Manual articulators include the place of the articulator, and hand configuration, movement, and orientation. Non-manual articulators consist of eye aperture, face, and body movement. Ideas can be fully expressed through the combination of both articulators.

The simultaneity of sign languages allows them to have similar rates of information transmission, despite of that a sign usually takes about twice as long to produce a word of spoken language (Bellugi & Fischer, 1972). Multiple visual cues are used in sign languages to convey different information simultaneously (CORMIER, 2006). For instance, facial expressions can modify lexical categories, a sentence can be negated by a head shake, and referents can be indicated by eye direction.

2.3 Written Representations for Sign Language

Sign languages do not have a standardized written form, and there are several notation systems and representations.

2.3.1 Visual Notation Systems

Since the 1960s, several notation systems were proposed, such as Stokoe notation (Kakumasu, 1968), SignWriting (Sutton, 1990), and HamNoSys (Prillwitz and Zinert, 1990). They mainly encode the position, movement, and orientation of the hands, or even the face and body as well into the form of symbols for SL linguistic analysis. Nonetheless, they represent only the visual nature of signs, the level but not the meaning of the signs.

2.3.2 Glosses as Meaning Representations

Glosses are typically used to represent the meaning of signs in the linguistic analysis of sign languages. A written gloss represents a sign in one or more words of a spoken language. For example, “CAR”, “Bridge”, and “CAR-CROSSES-BRIDGE”. They are produced by trained sign language linguists, thus their acquisition is very time-consuming and expensive. However, the meaning of signs cannot be represented accurately in all cases, as glossing has several limitations (Pfau et al., 2012):

- While signs often exhibit simultaneity, glosses are inherently sequential
- Glosses implicitly project the influence of the spoken language onto the sign language, as glosses are based on spoken languages.
- No universal standard on the construction of glosses

2.3.3 Sign Representations in Sign Language Processing

These sign representations are heavily used in sign language processing (SLP) research, which will be discussed in the next section. The visual notations are used as labels to pre-train feature extractors for research models, such as using SignWriting in the research of Koller et al. (2020). Glosses are often used as serving as labels for SLP research, due to the similarity of glosses to written language text.

2.4 Sign Language Processing (SLP)

Sign Language Processing (SLP) is an emerging area of artificial intelligence that focuses on the automatic processing and analysis of sign language content (Bragg et al., 2019). About a decade ago, due to the lack of Computer Vision (CV) technology in video processing, the early research efforts in Sign Language Processing were limited to using sensors to capture isolated signs and fingerspelling (Parton & Becky Sue, 2006). Later, the visual aspect of sign languages became the main focus of SLP research, leading to recent advances in CV. However, the current SLP techniques still fail to leverage or address the linguistic structure of SLs. Thus,

after the advancement of the CV techniques, much research is required to incorporate linguistic insight with them to achieve better sign language modeling and processing (K. Yin et al., 2021).

2.4.1 Common SLP Tasks and Their Limitations

According to Yin et al.(2021), the current common SLP tasks and their corresponding limitations as presented in Table 1:

SLP task	Aim
Detection	binary classification task to determine whether a sign is used in the given video
Identification	classification of identify which sign is used in a given video
Segmentation	detecting the frame boundaries of signs or phrases in the given video
Recognition (SLR)	determining the associated label (usually gloss) of each sign
Translation (SLT)	translation of sign language to spoken language
Production	producing sign language from spoken language

Table 1 - Current common SLP tasks

In this dissertation, we focus on the task of Sign Language Translation (SLT) for British Sign Language.

2.4.2 SL Recognition and SL Translation

Before the current heat research in SLP, SLR was the major focus of research and was one of the most difficult challenges to overcome(Núñez-Marcos et al., 2023). There are Isolated Sign Language Recognition (ISLR) and Continuous Sign Language Recognition (CSLR). The first one consists of classifying a sign performed in a cropped sign input such as a video of a single sign, while the second one deals with a stream of signs, aiming at segmenting and classifying them. The classification results are usually in glosses.

However, the gloss labeling of the dataset is costly. Also, the glosses from the CSLR model miss significant information from the sign language expression, not to mention producing natural and fluent spoken language text. CSLR also does not handle the syntax and

morphology of SL (Yin et al., 2021). Thus, Sign Language Translation (SLT) is one of the ultimate goals in SLP research. Although the gloss-free SLT is practically appealing, it introduces modeling challenges. One of the major challenges is that natural spoken language text or sentence provides far weaker supervision to the SLT modeling learning, as glosses are monotonically aligned to the signing(Shi et al., 2022).

With the development of technologies in Deep Learning, Computer Vision, and Machine Translation, the field of SLT is witnessing an increasing volume of research, which will be further discussed in the following sections. Figure 1 shows where SLT lie among different research area.

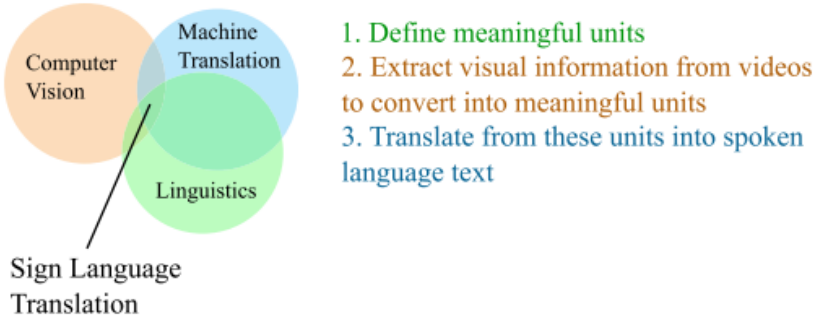

Figure 1 - SLT lies on the intersection of computer vision, machine translation, and linguistics(De Coster & Dambre, 2022)

2.5 Sign Language Machine Translation

Sign Language Machine Translation, or Sign Language Translation (SLT), is one of the common SLP tasks, and can be formulated as an input sequence of video frames that is transformed into a sequence of words. Since the sequence-to-sequence (seq2seq) formulation is widely adopted by Machine Translation (MT) for spoken languages (Tarrés et al., 2023), its development has extended to SLs. Sign MT was attempted to be used on SLP tasks including SL-to-text/speech (SLT) and speech/text-to-SL (SLP).

2.5.1 Development of Sign Language Machine Translation

Sign MT initially emerged through rule-based systems (Zhao et al., 2000) and subsequently to data-driven methods that relied on parallel corpora. In the early stages, the sign MT systems involved example-based translation and later progressed to statistical translation methods (Stein et al., 2012). These sign MT systems are also called Traditional Sign MT systems.

With the advancement in Deep Learning (DL) technology, which dominates NLP research (Young et al., 2017), there was a significant leap from the traditional sign MT systems to the current paradigm dominated by DL technology. DL-based sign MT, which is usually called Neural SLT is so promising that serves as the leading candidate for state-of-the-art technology in SLT (Núñez-Marcos et al., 2023).

Neural Networks allow translation to combine both the alignment and translation to and from multiple languages and even to create multilingual models. Compared to Traditional Sign MT, Neural SLT does not require looking for word alignment, nor creating ad-hoc rules for each individual language and so forth (K. Yin et al., 2021). Nonetheless, training neural network models necessitates a substantial volume of data, and the scarcity of available data acts as a hindrance to the research and development of Neural SLT.

2.5.2 Neural Sign Language Translation (NSLT)

The first methods in NSLT used Recurrent Neural Networks (RNNs) on an encoder-decoder architecture (Sutskever et al., 2014), with LSTMs or GRUs. However, there are limitations for RNNs to model long-term dependencies, especially for video input sequences captured with high frame rates. Later, attention-based methods were proposed to overcome the limitations by focusing on parts of the input during decoding (Bahdanau et al., 2014). After that, an end-to-end method using RNN encoder-decoder with attention, which is fed with 2D CNN visual features, was proposed by Camgoz et al.(2018).

Later, the Transformer neural networks emerged in the NMT field for NLP, which significantly improved the translation performance over the mentioned attention-based encoder-decoder approaches (Vaswani et al., 2017). It achieved good results in various other challenging NMT tasks, such as learning sentence representations, language modelling, and speech recognition. The Transformer relies on the self-attention mechanism that allows processing of the input sequence in parallel instead of sequentially. This enables better modelling for long-term dependencies and parallelization during training.

Since Transformer networks constitute the state-of-the-art paradigm for sequential data modelling, they were then applied to SLT tasks. The Transformer architecture was first attempted by Camgöz et al.(2020), and was proven to work well by various subsequent research (Tarrés et al., 2023). In this dissertation, some of the research working on Transformer networks is replicated and trained on datasets of British Sign Language.

2.6 Sign Language Corpora and Datasets

In general, there has been a large body of work in sign language corpora or dataset. In this dissertation, we focus on video-based corpora and datasets which can be used for SLT. They typically contain paired continuous signing videos and sentences in written language.

2.6.1 Sign Language Linguistic Corpora

Most of the early SL collections were targeted toward SL linguists instead of machine learning community. The SL linguistic collections are usually called SL corpus, and some of them are listed in Table 2. Nonetheless, the authors of the corpora have not prepared them for ease of use by the machine learning community (De Sisto et al., 2022).

Corpus	Lang	Format	Signers per file
ECHO Corpus (Nonhebel et al., 2004)	BSL	ELAN	1
BSL Corpus (Schembri et al., 2013)	BSL	ELAN	2
Corpus VGT (Van Herreweghe et al., 2015)	VGT	ELAN	2
Corpus NGT (Van Herreweghe et al., 2015)	VGT	ELAN	2
iSignos (Carmen Cabeza & Jos´e M. Garc´ıa-Miguel, 2018)	LSE	CSV	1

Table 2 - Examples of sign language corpora for linguistic use

ELAN is a software tool used in linguistics and anthropology for annotating sign videos. Annotations exported by this software are saved in an ELAN annotation format file (.eaf file) (Sloetjes Han & Wittenburg Peter, 2008).

2.6.2 Sign Language Machine Learning Datasets

With the progress of machine learning, specialized datasets have emerged explicitly designed for Sign Language (SL) machine translation or recognition tasks. However, in comparison to SL linguistic corpora, SL machine-learning datasets can present a potential drawback: they may contain non-authentic SL samples (De Sisto et al., 2022). For instance, some datasets are composed of TV broadcasts that have been augmented with SL interpretation, potentially featuring signers who are not members of the respective SL community but rather hearing SL interpreters. Table 3 shows a summary of current major sign language datasets, provided by De Sisto et al.(2022).

Sign Language Dataset	Lang	Vocab	Hours	Signers	Source

Phoenix-2014T (Camgoz et al., 2018)	DGS	3K	11	9	TV
KETI (Ko et al., 2019)	KSL	419	28	14	Lab
CSL Daily (Zhou et al., 2021)	CSL	2K	23	10	Lab
SWISSTXT-Weather (Camgoz et al., 2021a)	DSGS	1K	1	-	TV
SWISSTXT-News (Camgoz et al., 2021a)	DSGS	10K	10	-	TV
VRT-News (Camgoz et al., 2021a)	VGT	7K	9	-	TV
BOBSL (Albanie et al., 2021)	BSL	78K	1467	39	TV
How2Sign (Duarte et al., 2021)	ASL	16K	80	11	Lab
OpenASL (Shi et al., 2022)	ASL	33K	288	~220	Web

Table 3 - Examples of sign language machine learning datasets

2.6.3 Data Scarcity for SLT Research

Despite the large amount of work in SL data collections, there is still a scarcity of data for NSLT research. For the SL linguistic corpora, most of them have difficult accessibility. Also, they are collected in a studio-like and carefully controlled environment, thus they are hard to scale up and do not account for real-world conditions that are important for training a robust SLT model. The SL machine learning datasets suffer from problems of small vocabulary size, lack of visual variability, and being restricted to specific domains (Shi et al., 2022). Also, most existing SLT approaches rely on glosses, as building effective approaches for gloss-free SLT is still an under-studied challenge. As the annotation of SL videos with glosses is expensive, the data requirement of having glosses annotation further decreases the usable datasets for SLT training.

3 Related Work in SLT

Neural networks offer significant potential not only for continuous sign language recognition (CSLR) models, which translate sign videos into glosses but also for sign language translation (SLT) models. In this section, we will review related work on neural network architectures, with a particular focus on the Transformer model, in the context of SLT. Additionally, we will explore the datasets available for training SLT models, especially those tailored for British Sign Language. Furthermore, we will delve into the literature that discusses the extraction of features from raw sign videos, enhancing our understanding of this critical aspect.

3.1 SLT Tasks Classified by the Gloss Usage

Even with the neural network, an effective SLT model that directly translates signs to fluent spoken language text is yet to exist (Shi et al., 2022). Glosses recognized by the SLR system still play an important role in many SLT model training architectures, being the intermediate representation, training guidance, or supervision.

According to De Coster et al.(2023), distinct SLT tasks from various research can be classified according to whether, and how, glosses are used. The naming convention of the translation tasks is borrowed from Camgoz et al. (2020). These tasks include Gloss2Text, Sign2Gloss2Text, (Sign2Gloss, Gloss2Text), Sign2(Gloss+Text), and Sign2Text. They are illustrated in Figure 2.

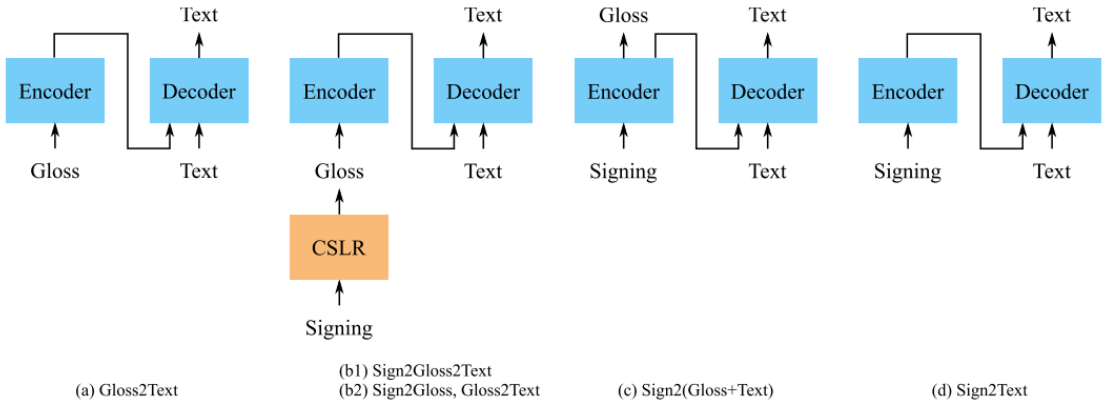

Figure 2 - Distinct translation tasks in the literature on SLT, classified by the involvement of gloss*(De Coster & Dambre, 2022)*

Before the approaches of using a Transformer neural network for the SLT task, which were first brought by Camgöz et al.(2020), most of the literature was either doing SLT as Sign2Gloss2Text or Sign2Text task. While direct Sign2Text models obtain only low performance, the mid-level gloss representation used by Sign2Gloss2Text methods introduces an information bottleneck, since:

(i) The gloss neglects a lot of information and crucial details in the original sign video, as it is an incomplete annotation that is intended only for linguistic study
(ii) The mid-level representation limits the network's ability to understand sign language but the gloss annotations instead.

Therefore, the next sections will explore how the research literature implements Transformer neural network architecture for the mentioned SLT tasks, with a particular focus on improving the utilization or possibly elimination of the gloss representation of signs.

3.2 Transformer Architecture on SLT Tasks

Proposed by Camgöz et al.(2020), it is the first Transformer architecture applied in SLT and constitutes the state-of-the-art approach at the moment. The Transformer networks train multiple co-dependent Transformer networks simultaneously for CSLR and SLT tasks in an end-to-end manner. Thus, the SLT task becomes a joint task Sign2(Gloss+Text), different from most of the previous literature using Signi2Gloss2Text, which is the previous state-of-the-art approach, or Sign2Text approaches. The new approach was an important breakthrough that spurred fresh research interest in the field. Many later SLT research works were built upon and extended the framework of Camgöz et al.(2020). Further details will be discussed in Section 4.2.1.

Nonetheless, Camgöz et al.(2020) mentioned that there are still drawbacks to using gloss supervision, especially the information bottleneck introduced by the usage of gloss, hindering the model from further understanding the language. Also, gloss annotating on SL

data is expensive, which is hard to perform for large-scale datasets. Later, there were research papers that proposed gloss-free Transformer approaches based on the Transformer approach of Camgöz et al.(2020).

In this dissertation, the Sign2(Gloss+Text) approach(Cihan Camgöz et al., 2020) and three gloss-free Transformer Sign2Text approaches(Tarrés et al., 2023; Voskou et al., 2021; A. Yin et al., 2023) are evaluated and reproduced.

3.3 Datasets for SLT Research

Section 2.6 has a general overview of the existing SL corpora and datasets. In this section, datasets that are currently used by most of the SLT research, and possible data collections that can be used for training the British Sign Language SLT model are reviewed.

3.3.1 PHOENIX-Weather 2014T Dataset

According to De Coster et al.(2023), the most used dataset for SLT research is RWTH-PHOENIX-Weather 2014T (Camgoz et al., 2018). It is the first dataset large enough for neural and is readily available for research purposes. The dataset contains sign videos in DGS, gloss annotations, and sentence text in German.

The dataset became a benchmark dataset in SLT research since its popularity facilitates the comparison of various SLT models trained with it. According to De Coster et al.(2023), the benchmark datasets are used by in total of 24 SLT research, within which there are 8 Gloss2Text, 5 Sign2Gloss2Text, 3 (Sign2Gloss, Gloss2Sign), and 8 Sign2(Gloss+Text) SLT method, while 8 of them are RNN and 16 are Transformer. The SLT research mentioned in Section 3.2 is all based on this dataset.

Nonetheless, PHOENIX-Weather 2014T contains only 11 hours of signing videos and 7096 training sentences. Compared to MT research of spoken languages with datasets typically containing several millions of sentences, sign language MT is a low-resource MT task (De

Coster et al., 2023). Also, the dataset is limited to a specific domain, which is weather forecast, and 9 signers.

3.3.2 Data of British Sign Language

As of the time of writing this paper, the BBC-Oxford British Sign Language Dataset (BOBSL) is the only machine learning dataset in BSL. BOBSL consists of 1467 hours of BBC BSL-interpreted TV broadcasts from 39 signers, comprising videos of continuous signing and subtitles corresponding to the audio content. The goal of the dataset is to provide a collection of BSL videos to support the research on SL recognition, alignment, and translation. According to Albanie et al. (2021), the advantages provided by BOBSL are:

(iii) Consists of co-articulated signs as opposed to isolated signs
(iv) Represents more natural signing (while still distinct from conversational signing due to its use of content interpretation)
(v) As the largest source of continuous signing (1467 hours)
(vi) Covers a large domain of discourse, genres, and topics
(vii) Automatically annotated more than 2,000 signs, though containing some noise

There are several research works on the automatic annotation of this dataset and its subset dataset BSL-1K, such as automatic sign spotting or annotation of individual signs and subtitle-sign alignment (Momeni et al., 2022), and sign video retrieval with textual queries(Duarte et al., 2021). The annotated signs and aligned subtitles are used for training the SLT model in this paper.

For the SL linguistics Corpus, British Sign Language (BSLCP) (Schembri et al., 2013) is a collection of BSL sign videos targeted toward SL linguists. The video data was collected from deaf native, near-native, and fluent signers across the UK, and recorded in a well-conditioned studio. The corpus provides descriptions of signing in ELAN. The dataset mainly focuses on manual articulators.

3.4 Sign Language Data Modalities

In most neural SLT architectures, SL videos are tokenized before they are fed into the neural network, like the Transformer encoder.

3.4.1 CNN Visual Feature Extraction Model

The first tokenization method is through a Convolution Neural Network (CNN). Camgoz et al. (2018, 2020) used a 2D CNN to extract features of sign video frames. The CNN was trained on another sign language recognition model (Zhou et al., 2021). Later, 3D CNNs were developed and have achieved state-of-the-art performance on many computer vision tasks, including sign language. One of the commonly used 3D CNNs is an inflated 3D convent (I3D), which "inflates" the 2D convolutional filters of Inception to 3D. It was originally developed for action recognition (Carreira & Zisserman, 2017) and further trained with SL data such as Samuel Albanie et al.(2020) and Duarte et al. (2021). In addition to the 3D CNN I3D, some SLT research uses 3D CNN S3D (Xie et al., 2017), which has more parameters and is more computationally intensive. Chen et al. (2022) use the S3D features, pretrained in kinetics and WLASL dataset.

3.4.2 Human Pose Estimator

The second tokenization method uses pose estimators (Cao et al., 2016) to represent the input sign video with information about the motion and position of body parts, which is particularly useful for action recognition. Poses in sign videos are extracted, normalized, and concatenated to form a video-level or frame-level processing. Mediapipe(Lugaresi et al., 2019) and OpenPose(Cao et al., 2021) are the most used pose estimators.

3.4.3 Other Modalities

There are other manually designed and sophisticated multi-cue channel tokenization methods, such as a two-stream network with raw frames and poses combined (K. Yin & Read, 2020), and spatial & and temporal multi-cue (hand, face, full-frame, and pose) networks (Zhou et al., 2020).

3.5 Summary

This section gives a general insight into the currently usable SLT approaches, BSL dataset, and modalities for SL that are possible to be applied to achieve the objective of this dissertation, which is to train SLT models for BSL. In the following section, details of the selected SLT approaches, BSL datasets, and SL modalities for the training experiments of this dissertation are explained.

4 Methodology

This section explains the details of the training of SLT models for BSL, including the BSL data, model training approaches, and evaluation metrics.

4.1 BSL Data

Two BSL datasets are used for training the SLT models in this paper, including the linguistic corpus BSLCP (Schembri et al., 2013) and the large-scale interpretation-based dataset BOBSL (Albanie et al., 2021). The following subsections explain how the datasets were requested to access and their usable data, while the details of data preparation, including rearranging and pre-processing for the SLT model training, are explained in Section 5.

4.1.1 BOBSL Data

As mentioned in Section 3.3.2, it is only permitted to use BOBSL for academic research related to the translation of BSL. After getting approval from the BBC on the "BBC BOBSL Terms of Use" agreement, access to the BBC R&D website was obtained. Figure 3 shows the structure of the dataset BOBSL. The folders or files that are mainly used in the experiment of this paper include 'videos', 'features/i3d_c2281_16f_m8_-15_4_d0.8_-3_22', 'subtitles/manually-aligned'.

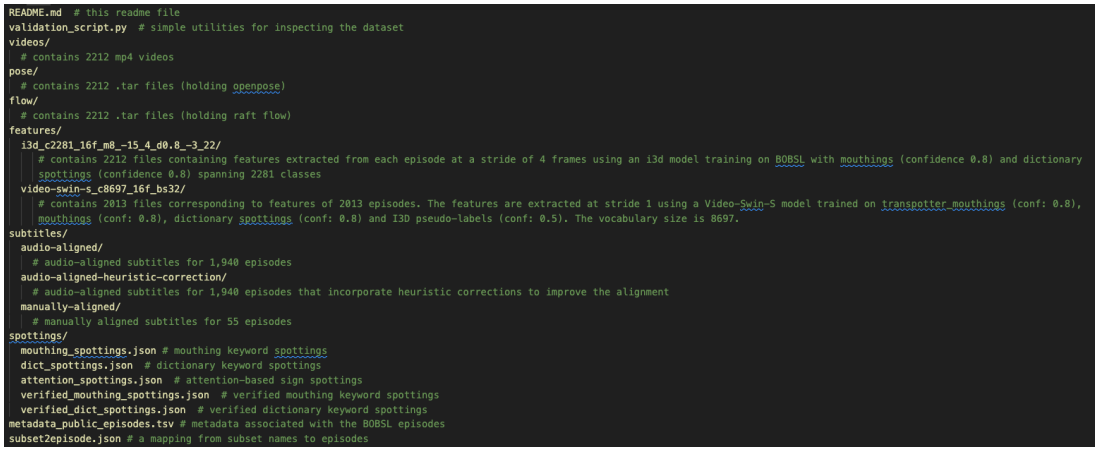

Figure 3 - Structure of dataset BOBSL

Usable Sentence-level Data

The training of SLT models with Transformer architectures mentioned in Section 4.2 requires sign language data with sentence-level annotation. Therefore, sign interpretation videos are usable for training only if their subtitles are aligned with the signing. Figure 4 shows an example of subtitle alignment. Although BOBSL consists of 1940 episodes (about 1467 hours) of BBC broadcasts with subtitles and sign interpretation (Albanie et al., 2021), their subtitles are by default coarsely synchronized with the audio track. Thus, only 55 episodes (about 50 hours) with sign-aligned subtitles, which are provided by the internal dataset BSL-1K of BOBSL, were pre-processed and used for training.

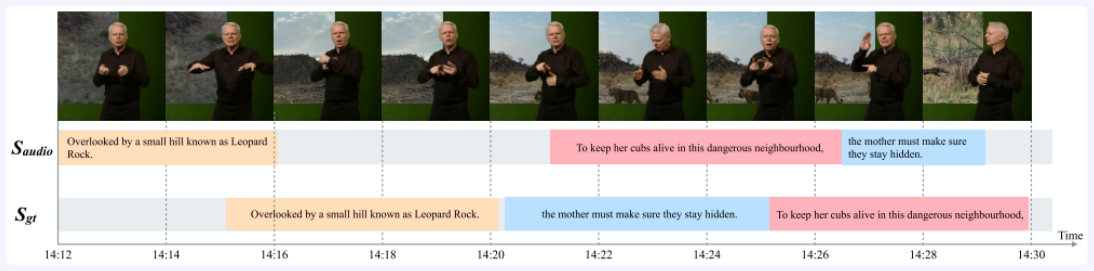

Figure 4 - Subtitle alignment (Bull et al., 2021)

Sign-aligned subtitles format

In the 'subtitles/manually-aligned' folder, there is one '.vtt' file for manually aligned subtitles per one episode. For example, file '5085344787448740525.vtt' is the sign-aligned subtitle file of BBC episode ID '5085344787448740525'. Figure 5 shows how the subtitles are organized corresponding to the aligned time of the signing. The time range of each subtitle sentence is denoted, which indicates when the sign interpretation of that spoken language sentence was performed.

```
00:05:18.827 --> 00:05:20.745
Progress.

00:05:21.785 --> 00:05:24.021
So they say.

00:05:26.017 --> 00:05:40.763
The forces of change have decided that the tower's life as a council block is over with all but one of its residents now removed to inner city estates

00:05:43.597 --> 00:05:51.271
But Les Brookes, a professional clown, bought his flat off the council and is refusing to sell it back.

00:05:51.271 --> 00:05:55.065
I bought the flat because I wanted to try and stay here.

00:05:55.065 --> 00:06:11.610
In negotiations with the council over the past 18-odd months, they've been asking me to move out, sending semi-threatening letters, some more threater

00:06:11.610 --> 00:06:18.811
And... over the course of the last year, the heating has been turned off...

00:06:18.811 --> 00:06:20.678
It's become a fight now.
```

Figure 5 - File of the sign-aligned subtitle of one BBC episode

4.1.2 BSLCP Data

Upon the approval of the request access to the BSLCP restricted access data, access to sign language video of conversation and interview with annotations were obtained. The data is accessed through the website of UCL Library, in which there are 454 separate files that were downloaded one by one. Figure 6 shows the interface of the library website.

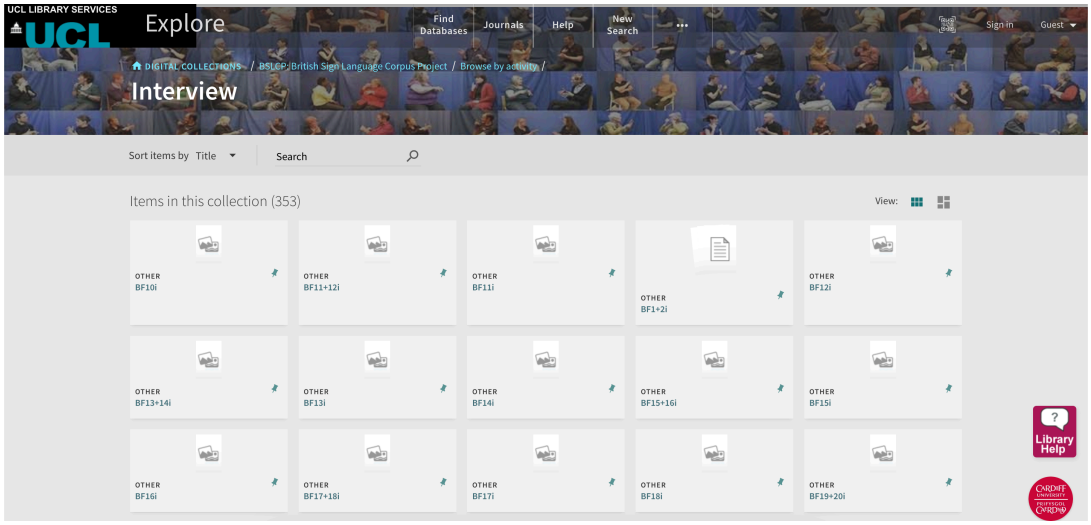

Figure 6 - UCL library website for BSLCP

Different from BOBSL, BSLCP has linguistic annotations that are translated into spoken language sentences. They are already aligned to the signing and are stored in ".eaf" files. ".eaf" is a file format used by ELAN annotation software. Nonetheless, only a portion of videos have their corresponding translation files. There are 237 sign language videos, which is about 80 hours, downloaded with their ".eaf" translation files. Figure 7 shows a part of

the ".eaf" file, in which there are two spoken language sentence annotations with specified starting and ending times.

```
<TIME_ORDER>
    <TIME_SLOT TIME_SLOT_ID="ts1" TIME_VALUE="7580"/>
    <TIME_SLOT TIME_SLOT_ID="ts2" TIME_VALUE="14730"/>
    <TIME_SLOT TIME_SLOT_ID="ts3" TIME_VALUE="14900"/>
    <TIME_SLOT TIME_SLOT_ID="ts4" TIME_VALUE="19580"/>
    ...
</TIME_ORDER>
<TIER DEFAULT_LOCALE="en" LINGUISTIC_TYPE_REF="BasicAnnotation" TIER_ID="FreeTransl">
    <ANNOTATION>
        <ALIGNABLE_ANNOTATION ANNOTATION_ID="a1" TIME_SLOT_REF1="ts1" TIME_SLOT_REF2="ts2">
            <ANNOTATION_VALUE>Signing in BSL means that there is no word-order to follow, at all.</ANNOTATION_VALUE>
        </ALIGNABLE_ANNOTATION>
    </ANNOTATION>
    <ANNOTATION>
        <ALIGNABLE_ANNOTATION ANNOTATION_ID="a2" TIME_SLOT_REF1="ts3" TIME_SLOT_REF2="ts4">
            <ANNOTATION_VALUE>You say everything when signing, but not in that order.</ANNOTATION_VALUE>
        </ALIGNABLE_ANNOTATION>
    </ANNOTATION>
```

Figure 7 - ELAN annotation ".eaf" file

4.1.3 Usable Data Summary

Dataset	Videos	Length	Sentences	Words
BOBSL	55 (episodes)	37.7 hrs	31479	337248
BSLCP	237	34.5 hrs	28853	207093

Table 4 - Summary of all of the usable BSL data

4.2 Model Training Approaches

This section introduces the four different approaches for the SLT model training (Camgöz et al., 2020; Tarrés et al., 2023; Voskou et al., 2021; A. Yin et al., 2023). One of the major reasons for selecting them is that they publish the code online, making it possible to reproduce the SLT model training, especially on BSL datasets. This section explains the reasons for selecting and their architectures. The four SLT approaches with their corresponding papers are listed in Table 5.

Approaches	Title of SLT Research Papers	Abbreviation
Sign-to-(Gloss+Text)	Sign Language Transformers: Joint End-to-end Sing Language Recognition and Translation (Camgöz et al., 2020)	S2(G+T)-Trans
Sign-to-Text	Stochastic Transformer Networks with Linear Competing Units: Application to end-to-end Sign Language Translation (Voskou et al., 2021)	S2T-LCU-Trans
Sign-to-Text	Gloss Attention for Gloss-free Sign Language Translation (A. Yin et al., 2023)	S2T-GA-Trans
Sign-to-Text	Sign Language Translation from Instructional Videos (Tarrés et al., 2023)	S2T-Trans

Table 5 - SLT approaches with their corresponding research papers

Each SLT research paper is assigned an abbreviation for ease of discussion in the following sections. S2(G+T) and S2T means the SLT approaches Sign-to-(Gloss+Text) and Sign-to-Text, respectively, while "Trans" stands for Transformer. LCU and GAL refer to the major techniques its corresponding paper proposed, which are Linear Competing Units and Gloss Attention respectively.

4.2.1 S2(G+T)-Trans: Transformers for joint end-to-end SLR and SLT

Proposed by Camgöz et al.(2020), this paper is the first paper to apply a Transformer network to SLT tasks and became the state-of-the-art SLT approach. The Transformers jointly learn to recognize and translate sign video sequences into sign glosses and spoken language sentences. In other words, the SLT method of their paper is a Sign2(Gloss+Text) task. With this Transformer Architecture, they managed to achieve the highest BLEU-4 score, which is 21.80, on dataset PHOENIX-2014T (Camgoz et al., 2018) at that time.

Technical Objectives

According to Camgöz et al.(2020), there are two reasons for the previous Sign2Text approach achieves low performance. First, the number of sign glosses is a lot lower than that of the sign video frames, resulting in a long-term dependency issue. Second, the Sign2Text approach lacks direct guidance for understanding sign sentences during the training. Sign2(Gloss+Text), rather than the Sign2Text approach, avoids the long-term dependencies issue and provides gloss supervision.

For the reasons of Sign2(Gloss+Text) over the Sign2Gloss2Text, previous state-of-the-art approach, they considered that the use of glosses as intermediate representations could be harmful. There is an information bottleneck introduced by the mid-level gloss representation. The gloss has an inherent loss of information as it is an incomplete annotation, it also limits the network's ability to understand sign language. Thus, instead of

being the intermediate representations, glosses in the proposed multi-tasking strategy only provide the learning guidance.

Embedding Layers

For the word embedding, the one-hot-vector representations of spoken language words are projected into embedded representations, which is a denser space, through a linear layer. The linear layer was initialized from scratch during training.

For the spatial embedding, they use the method proposed in the previous paper of Camgoz et al.(2018), frame images of sign videos (2D array of pixel values) are propagated through a pre-trained 2D CNN(Koller et al., 2020), and transformed into a non-linear spatial representation as a denser vector.

Transformer Architecture

The Sign Language Transformers network consists of a Sign Language Recognition Transformer (SLRT) encoder and an autoregressive Sign Language Translation Transformer (SLTT) decoder. They will be explained in the following sections respectively. Figure 8 shows the SL Transformers architecture.

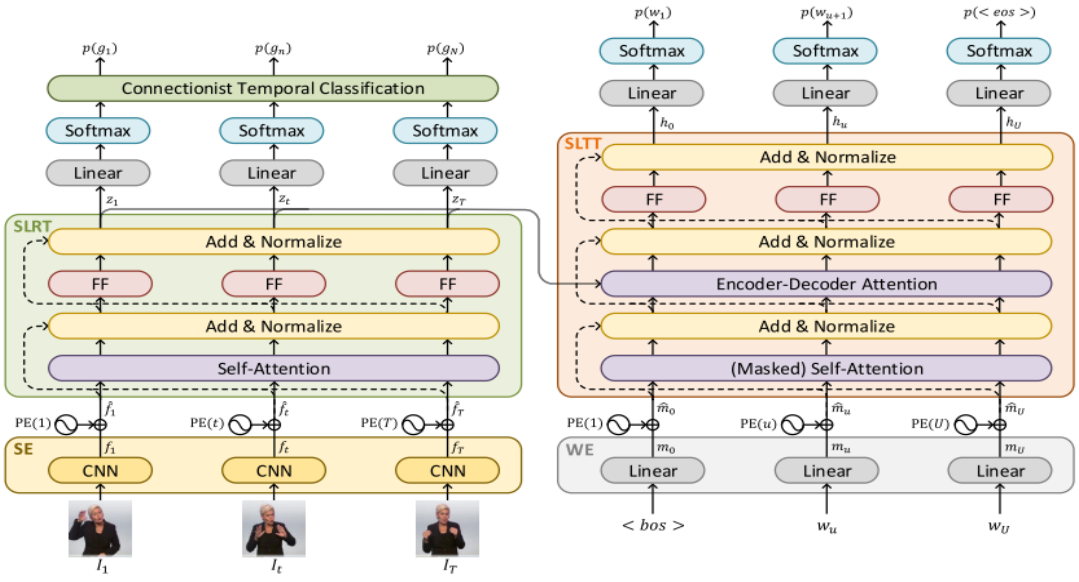

Figure 8 - A detailed overview of a single-layered Sign Language Transformer (Camgöz et al., 2020)

(SE: Spatial Embedding, WE: Word Embedding, PE: Positional Embedding, FF: Feed Forward)

For the SLRT encoder, the primary function is to (i) learn meaningful spatiotemporal representations without limiting the information being passed to the SLTT decoder and (ii) recognize intermediate glosses from the SL video for the supervision of learning.

For the SLTT decoder, it is an Autoregressive Transformer Decoder. The primary function of the Decoder is to generate spoken language sentences by exploiting the spatiotemporal representations learned by the SLRT Encoder. The SLTT Decoder consists of a self-attention layer and an encoder-decoder attention module.

Loss Calculation

For the SLRT encoder, Camgöz et al.(2020) did not use cross-entropy loss with frame-level annotations, since it is hard for sign gloss annotations to have such precision. Instead, they use a sequence-to-sequence learning loss function, namely CTC, which provides weaker supervision. The loss calculation can be formulated as:

$$\mathcal{L}_{\mathcal{R}} = 1 - \sum_{\pi \in \mathcal{B}} \mathrm{p}(\,\pi \mid \mathcal{V}\,)$$

For the SLTT decoder, it is obtained by calculating the cross-entropy loss for each word, which can be formulated as:

$$\mathcal{L}_{\mathcal{T}} = 1 - \prod_{\mathrm{u}=1}^{\mathrm{U}} \sum_{\mathrm{d}=1}^{\mathrm{D}} \mathrm{p}\big(\widehat{\mathrm{w}_{\mathrm{u}}^{\mathrm{d}}}\big)\mathrm{p}\big(\,\mathrm{w}_{\mathrm{u}}^{\mathrm{d}} \mid \mathrm{h}_{\mathrm{u}}\,\big)$$

The whole Transformer network is trained by minimizing the joint loss term $\mathcal{L}$, which is the weighted sum of the recognition loss $\mathcal{L}_{\mathcal{R}}$ from the SLRT Encoder and the translation loss $\mathcal{L}_{\mathcal{T}}$ from the SLTT Decoder. This process can be represented as:

$$\mathcal{L} = \lambda_{\mathrm{R}}\mathcal{L}_{\mathcal{R}} + \lambda_{\mathrm{T}}\mathcal{L}_{\mathcal{T}}$$

4.2.2 S2T-LCU-Trans: Stochastic Transformer with LCUs

The second paper by (Voskou et al., 2021) introduces an approach that treats SLT as a Sign2Text task. Their proposed approach enables the trained SLT model to achieve the currently highest BLEU-4 score on PHOENIX-14T, which is 25.59(Tarrés et al., 2023).

For the embeddings of the video and text, they use the same way as the previous paper of Camgöz et al.(2020), including the video feature extractor and the spatial and word embedding layers.

Technical Objectives

They were aware of the problem that most of the previous approaches that yielded viable SLT performance, like the Sign2(Gloss+Text) approach by Camgöz et al.(2020) mentioned in the previous Section 4.2.1, require laborious gloss sequence groundtruth. The goal of this paper is to build a gloss-free method training for the Transformer network to do direct Sign2Text SLT while achieving a similar or better performance than the previous state-of-the-art Sign2(Gloss+Text) on dataset PHOENIX-2014t.

Transformer Architecture

The Stochastic Transformer Architecture proposed by Voskou et al.(2021) was largely extended upon the architecture of the Sign2Text method of (Camgöz et al., 2020), which was also considered as their baseline. They proposed a novel type of layers that combines: (i) local winner-takes-all (LWTA) layers with stochastic winner sampling instead of ReLU layers, (ii) stochastic weights with posterior distributions, and (iii) a weight compression technique at inference time. A comparison of the Transformer Architecture overview is shown in Figure 9.

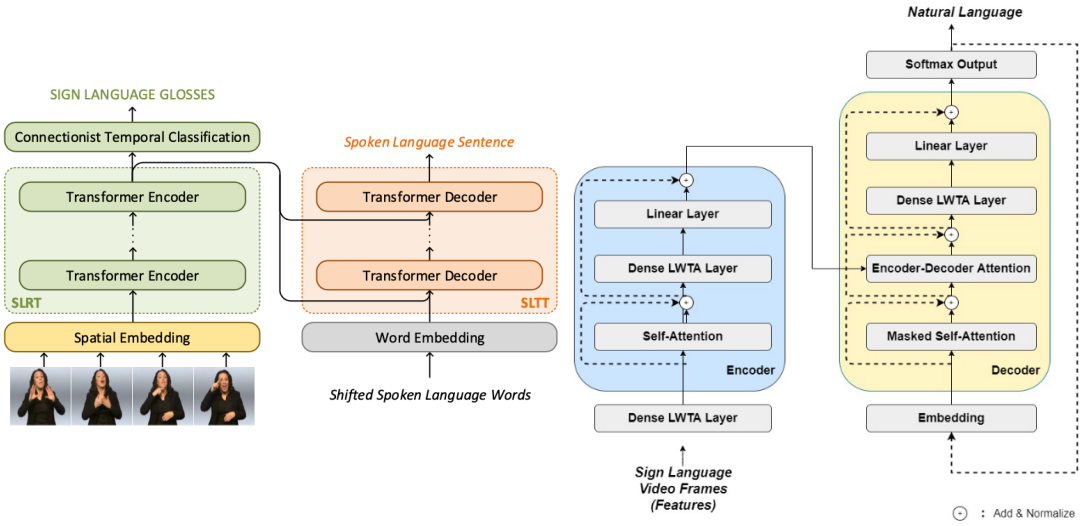

Figure 9 - Comparison of Transformer Architectures

(a) Left - Transformer of Sign2(G+T) (section 4.2.1) (Camgöz et al., 2020)

(b) Right - Stochastic Transformer of S2T-LCU-Trans (Voskou et al., 2021)

For the Stochastic Winner-Takes-All (LWTA) Layers, it is to replace the dense ReLU layers are used in the conventional Transformer by introducing LWTA layers, which make use of Linear Competing Units (LCUs) with Stochastic Winner-Take-All (WTA) activations in the network. They mentioned that the LWTA layers allow for more efficient and effective competition between neurons, which improves the model's ability to capture complex patterns in the data. Additionally, the authors use a variational Bayes approach to fit Gaussian posteriors over the weights of the LCUs, which allows for more flexible and robust modelling of the weight distributions.

Loss calculation

The standard categorical cross-entropy error is used for training in conventional Transformer networks, in which the output is assumed to be a deterministic function of the input. In terms of the STN, the model includes LCUs with WTA activations, which introduce additional sources of uncertainty. Therefore, instead of reducing the cross-entropy error, the STN training is to maximize the evidence lower-bound (ELBO). ELBO is a lower bound on the log-likelihood of the data, which takes into account the uncertainty while capturing the underlying structure of the data.

4.2.3 SLT-GA-Trans: Gloss-free Transformer with Gloss Attention

The third paper (A. Yin et al., 2023) proposed an SLT network called Gloss Attention SLT (GASLT) which is actually a gloss-free Sign2Text method. They first also did a throughout analysis of how gloss annotations make SLT easier in Sign2(Gloss+Text) task. Then, they modified the Transformer architecture by proposing a gloss attention mechanism to replace the self-attention mechanism which is original in the encoder, and introduced the usage of knowledge transfer from natural language models to the SLT model to enhance the model's capability in capturing global information in the sign language videos. They claimed that the model has a better understanding of the sign language videos at the sentence level.

Embedding Layers

For the spatial embedding, A. Yin et al.(2023) first extract the feature using a pre-trained 3D CNN I3D model from the TSPNet architecture, which is different from the previous two papers which use a 2D CNN model. Then, similar to the previous two papers, a linear layer is used to convert the features to a higher-dimensional space, which is the same as the dimension of the encoder, followed by batch normalization with a ReLU activation function. Last, positional information is injected using positional encoding.

For the text embedding, they first use a BPE (Sennrich et al., 2015) sub-word segmentation model called BPEmb (Heinzerling & Strube, 2017) to segment text into sub-words. The BPE model was learned from the Wikipedia dataset using the SentencePiece (Kudo & Richardson, 2018). This is helpful for low-frequency word learning and out-of-vocabulary problems, as it allows for generalized phonetic variants or compound words(A. Yin et al., 2023). After that, the pre-trained sub-word embeddings in BPEmb are used as the initialization of the embedding layer.

Original self-attention and proposed gloss attention

For the original self-attention, according to A. Yin et al.(2023), without the supervision of the gloss annotation, there are two problems with this calculation: (i) the computational complexity is quadratic, and (ii) attention is difficult to converge to the correct position. The reason is that each query has to calculate the attention score with all keys. This approach can only be effective and flexible when strong supervision information is provided, otherwise, the model loses focus.

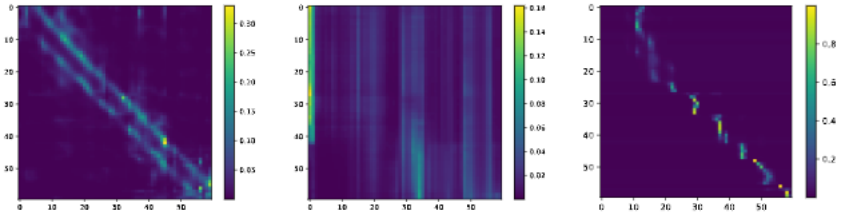

Figure 10 - Visualization of the attention map (A. Yin et al., 2023)*, (a) gloss provides alignment information for the model, (b) traditional attention calculation without the gloss supervision signal, (c) still flexibly maintains the attention in important regions*

The proposed gloss attention was designed by A. Yin et al.(2023) according to the characteristics of sign language. They observed that gloss-level semantics are temporally localized, as visualized in Figure 10a. It is likely to be in the same semantic for the adjacent video frames. Considering this, they first initialize N attention positions for each query. To deal with the semantic boundary problem, the position of the attention is dynamically adjusted by calculating the N offset according to the input query. With the adjusted attention position, the keys and values are obtained for the attention score calculation.

The result of the paper shows that the gloss attention they designed achieved similar results with self-attention with gloss supervision, which allows the model to keep the attention in the correct position. Also, the computational complexity, O(n), is better than self-attention which has quadratic complexity if there is no supervision information. Figure 11 shows the full gloss attention operation.

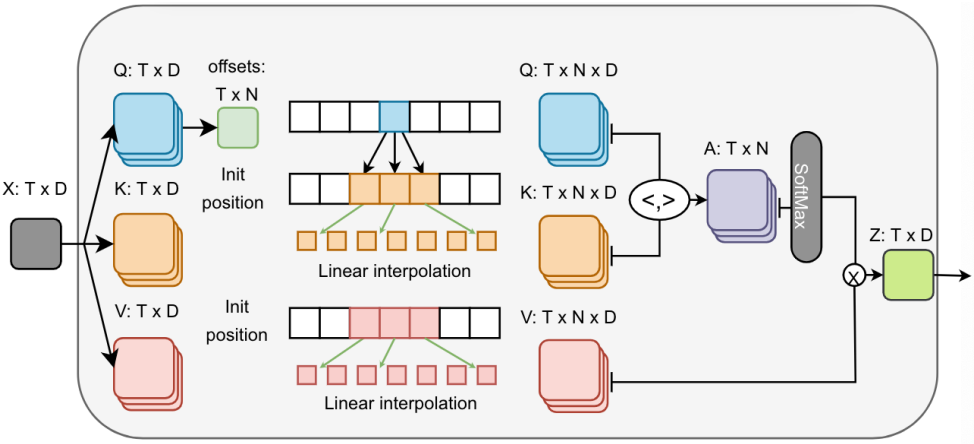

Figure 11 - Gloss attention flowchart (A. Yin et al., 2023)

Global information

As mentioned by A. Yin et al.(2023), another important role of gloss in the original Sign2(Gloss+Text) SLT model training architecture is to help the model understand the entire sign video from a global perspective. The ability of the model to capture global information decreases without the gloss information. Concerning this issue, they transfer knowledge from the language model that was learned on a rich corpus of natural languages to the SLT model, using the one-to-one semantic relationship between sign language video and annotated natural language text.

To transfer the knowledge, they first use sentence BERT (Reimers & Gurevych, 2019) to calculate the cosine similarity between all of the natural language sentences. After that, they aggregate the video features output to obtain an embedding. With the cosine similarity and the embedding vector, the knowledge transfer is obtained by minimizing the mean squared error of cosine similarity between video vectors and cosine similarity between natural language sentences. In this way, the model knows which sign language videos are linguistically similar and which are semantically different, helping the model to recognize patterns and relationships between the signs and gestures in videos.

4.2.4 S2T-Trans: Conventional Transformer with Regularization

For the forth paper “Sign Language Translation from Instructional Videos”, Tarrés et al.(2023) utilize a Sign2Text approach and a standard Transformer architecture with conventional encoders and decoders without any modification, which is different from the previously mentioned architectures, namely S2T-LCU-Trans and S2T-GA-Trans.

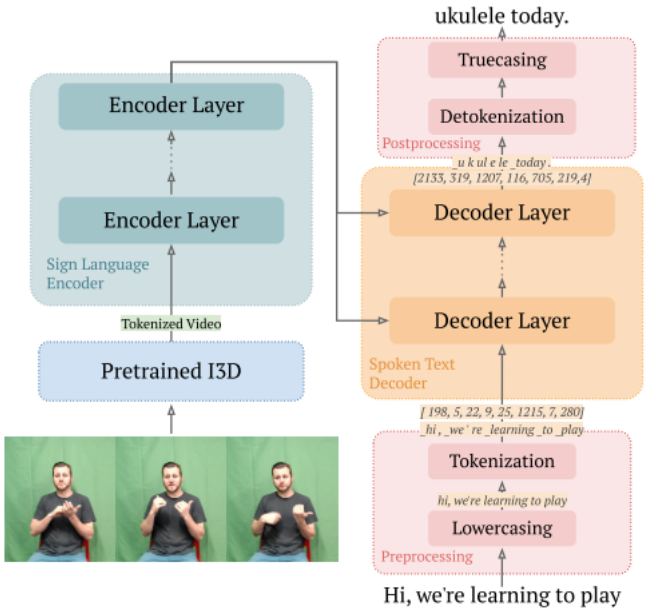

Figure 12 - Overview of the Transformer model training of S2T-Trans (Tarrés et al., 2023)

Network training regularizations

Although Tarrés et al.(2023) did not modify the Transformer, they added regularization during the training to prevent the model from overfitting. They mentioned that the regularization enables a larger model with a deeper Transformer network, resulting in better inference performance of the trained model. The regularization includes adding dropout, weight decay, and label smoothing, which will be further discussed in the Section 7 along with the experiment results of this paper.

Embedding Layers

For the tokenization or embedding of the video, Tarrés et al.(2023) use an I3D model, which was pre-trained on the ImageNet dataset and fine-tuned on the Kinetic-400 dataset and

sign language data through training for isolated sign recognition task, to extract features from video frames.

For the text tokenization, they convert the raw text to lowercase and employ Sentencepiece tokenizer with various vocabulary sizes to segment the lowercase text into sub-word units. Sub-word tokenization allows better handling of unseen words, that can be represented as combinations of sub-words from the vocabulary.

4.3 Evaluation Metric

As the most common metric for machine translation, the bilingual evaluation understudy score (BLEU) (Papineni et al., 2002) is utilized to measure the translation performance of the SLT models trained using various Transformer Architectures mentioned in the previous section. Multiplied by 100 in many cases, its value is always between 0 to 100, indicating how similar the translated text is with respect to the annotations or references. The calculation of the BLEU scores for evaluating the SLT model is explained as follows.

4.3.1 N-gram

N-gram is a contiguous sequence of n items from a given sequence of text. To calculate the BLEU scores, N-grams are used in the BLEU metric to measure the overlap or count the number of matches between the candidate translation (predicted text from the SLT model) and the reference translation in terms of the sequences of words they contain.

4.3.2 Modified N-Gram Precision

Nonetheless, the matches of N-grams are not directly used for calculating the BLEU. Modified N-gram Precision is applied to avoid situations of the model overgenerating “reasonable” words, resulting in a translation that is improbable but high-precision. The idea behind the Modified N-gram Precision is that a word/n-gram in the reference translation should be considered exhausted after a matching word in the predicted translation is identified(Papineni et al., 2002).

Predicted/Candidate Sentence	The the the the the the the
Reference Sentence	The cat is on the mat
Modified N-gram Precision Scores	2/7

Table 6 - Example of the Modified N-gram Precision Scores of a predicted-reference sentence pair

To calculate the Modified N-gram Precision, the number of times that a word/n-gram occurs in the reference translation clips as the total count of matched word/n-gram. Then, The clipped counts are added up and divided by the total unclipped number of predicted words. Table 6 shows an example of the Modified N-gram Precision. The calculation of the Modified N-gram Precision score, p_n, can be formulated as:

$$p_n = \frac{\sum_{n\_gram \in C} Count_{clip}(n\_gram)}{\sum_{n\_gram' \in C'} Count(n\_gram')}$$

Papineni et al.(2002) mentioned that predicted translation using the same words/1-grams as the references tends to satisfy adequacy, while longer n-gram matches account for fluency.

4.3.3 Brevity Penalty (BP)

Although predicted/candidate sentences that are either long or short are penalized through penalizing spurious words that do not appear in the reference translation, some sentence that is absurdly short compared to the proper length, such as “of the”, still inflates the precision. With the Brevity Penalty, a high-scoring predicted/candidate translation must match the length of the reference translations, in word/n-gram choice and order. The Brevity Penalty can be shown as:

$$BP = \begin{cases} 1, & \text{if } c > r \\ e^{(1-r/c)}, & \text{if } c \leq r \end{cases}$$

where c is the candidate translation length, and r is the effective reference corpus length.

4.3.4 BLEU Calculation

The BLEU scores are calculated through the product of the brevity penalty factor and the exponential of the weighted sum of the logarithms of the modified precision scores, which can be formulated as:

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

where

- p_n is the Modified N-gram Precision
- Logarithm function used to dampen the effect of outlier
- w_n is the weight that determines the relative importance of the different n-gram orders
- N is the maximum n-gram length
- Exp is the exponential function for converting the logarithmic scale of the modified precision scores back to a linear scale, thus the final BLEU score is between 0 and 1

4.3.5 BLEU for SLT Model Evaluation

The computation of BLEU values employs a cumulative weighting of n-grams of sizes rising from 1 to 4, being BLEU-1, BLEU-2, BLEU-3, and BLEU-4, where BLEU-4 is known as the BLEU scores. For instance, the formula of BLEU-1 is the formula shown in the previous section in which the N is 1, while the formula of BLEU-4 is the same in which the N is 4.

4.4 Summary

This section has explained the BSL datasets, Transformer training approaches, and the evaluation metrics that are used in the experiments of this dissertation. The BSL datasets include the linguistic corpus BSLCP and the interpretation-based dataset BOBSL. As shown in Table 7, the training implementations of approaches proposed by four papers are reproduced on BSL datasets with some modifications, mainly regarding the visual feature extraction or

pre-processing of SL videos. BLEU is the evaluation metric of the performance of the trained SLT models.

Training Approach	Pre-processing		Transformer Modifications	Encoders/ Decoders	Training Loss Function
	Visual	Text			
S2(G+T)-Trans (SLR+T)	2D CNN	-	SLRT encoder (glosses for supervision) SLTT decoder	3-3	SLR: Connectionist Temporal Classification (CTC) SLT: Cross-Entropy Loss
S2T-LCU-Trans (SLT)	2D CNN	-	Linear Competing Units Stochastic Winner-Takes-ALL layers	3-3	Evidence Lower-Bound (ELBO) Maximization
S2T-GA-Trans (SLT)	I3D CNN	BPEmb	Gloss Attention Global Information (transfer knowledge)	2-2	Cross-Entropy Loss
S2T-Trans (SLT)	I3D CNN	Senten cepiece	Regularizations prevent overfit (dropout, weight decay, label smooth)	6-3	Cross-Entropy Loss

Table 7 - Summary of training implementations of the four papers

5 Experiments

This section explains the details of data preprocessing and summarizes the implementation of the SLT model training.

5.1 Datasets Pre-Processing

Data of BSL Datasets, including BOBSL and BSLCP, are rearranged, reformatted, and pre-processed according to the data used for the original SLT Transformer training of the four previous papers mentioned in Section 4.2. The following sub-sections include the original data format and arrangement, sentence-level data rearrangement, extracted sign language video features, and sentence tokenization and similarity.

5.1.1 Original Data Format and Arrangement

The papers of S2(G+T)-Trans (Section 4.2.1), S2T-LCU-Trans (Section 4.2.2), and S2T-GA-Trans (Section 4.2.3) use the same dataset, PHOENIX-2014T. Thus, the data format and arrangement are the same for them, though the gloss information included in the data is only used by the training of the S2(G+T)-Trans. For the S2T-Trans, the dataset How2Sign that they used is arranged differently than that of the first three papers using PHOENIX-2014T. To replicate the SLT model training using the Transformer architectures of the four papers, data of BSL datasets is prepared and rearranged according to how the data of datasets PHOENIX-2014T and How2Sign are organized for the paper's original training experiments.

5.1.1.1 Format of PHOENIX-2014T Data

As mentioned in Section 3.3.1, the GSL dataset PHOENIX-2014T has been a benchmark dataset in SLT research since its popularity facilitates the comparison of various SLT models trained with it. The S2(G+T)-Trans, S2T-LCU-Trans, and S2T-GA-Trans experimented on this dataset with the same format. There are three data files including

“phoenix14t.pami0.train”, “phoenix14t.pami0.dev”, and “phoenix14t.pami0.test”. The files are binary files that were compressed and serialized by, possibly, using gzip and the Python pickle module respectively. After each data file is decompressed and deserialized, it becomes an object that contains the portion of data, train, dev, or test. The resulting object is formatted as a list, with each element representing a Python dictionary containing data for a sign language sentence-level entry.

For the content of the Python dictionary of each sentence-level data entry, it contains keys including “name”, “signer”, “gloss”, “text”, and “sign”. An example of a sentence-level data entry is shown in Figure 13.

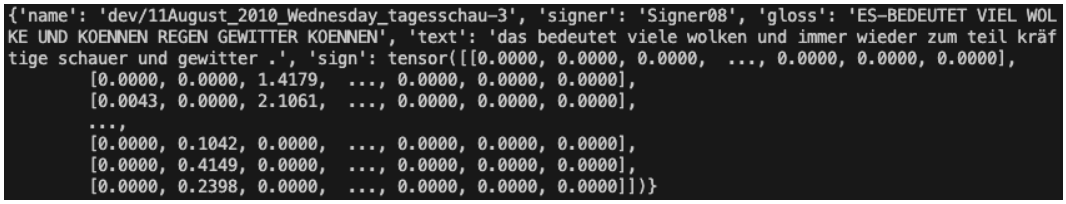

Figure 13 - Example of the dictionary of a sign language sentence-level data entry

Sign Language Video representations

The data files do not contain sign language videos, or frame images cropped from the videos. Instead, they are sign or video features which are frame representations that are already extracted by a 2D CNN (Camgöz et al., 2020). The 2D CNN was an Inception network(Szegedy et al., 2017) that was pre-trained for sign language recognition in a CNN+LSTM+HMM setup (Koller et al., 2020). In the setup, they opted for the 22-layer deep GoogleNet architecture (Szegedy et al., 2014), which is a kind of Inception network. The sign videos were cut into frames with 25 frames per second. Then each frame passes through the 2D CNN features extractor to get the frame representations. The frame representations of the video features extracted by the 2D CNN is N*1024, where N is the number of the frames of the video corresponding to that sign language sentence.

5.1.1.2 Format of How2Sign Data

The ASL dataset How2Sign is introduced by the paper of Tarrés et al.(2023). The experiment of S2T-Trans (Section 4.2.4) was trained on How2Sign to provide the first baseline result for the dataset. The data used for the baseline training experiments include a folder consisting of Numpy files of I3D features of the video section corresponding to the sign sentence, and three TSV files, including “train.tsv”, “val.tsv”, and “test.tsv”, each one consists of the information of the sign sentences. As an example, Figure 14. shows several rows of a TSV file.

id	signs_file	signs_offset	signs_length	signs_type	signs_lang	translation
-d5dN54tH2E_0-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_0-1-rgb_front.npy	324	165	i3d	asl	We're going to work on a arm drill that will help you have graceful hand movements in front of you.
-d5dN54tH2E_1-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_1-1-rgb_front.npy	516	92	i3d	asl	I call it painting the wall.
-d5dN54tH2E_10-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_10-1-rgb_front.npy	2137	59	i3d	asl	So we're going to go up and down; let's switch hands, down and up; down and up.
-d5dN54tH2E_11-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_11-1-rgb_front.npy	2196	145	i3d	asl	And just let those fingers relax.
-d5dN54tH2E_12-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_12-1-rgb_front.npy	2356	179	i3d	asl	Now together you're going to go opposite.
-d5dN54tH2E_13-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_13-1-rgb_front.npy	2553	199	i3d	asl	Do it very mechanically at first and then slowly soften it out as you feel comfortable.
-d5dN54tH2E_14-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_14-1-rgb_front.npy	2797	104	i3d	asl	It's real easy to actually get your fingers to lead, so try not to let them do that.
-d5dN54tH2E_15-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_15-1-rgb_front.npy	2900	121	i3d	asl	Let the wrist do all the leading.
-d5dN54tH2E_2-1-rgb_front	/path_to_features/how2sign/i3d_features/val/-d5dN54tH2E_2-1-rgb_front.npy	650	275	i3d	asl	So this one what you're going to start with, let's take one hand first; it's going to, the wrist is going t

Figure 14 – Example of data rows in the TSV file

Sign Language Video representations

The video representations for each sign language sentence are stored as Numpy files that consist of a tensor of shape frames*1024. They were extracted using a fine-tuned I3D network, which is a 3D CNN. The I3D network was pre-trained on ImageNet data(Jia Deng et al., 2009) and fine-tuned for action recognition with the Kinetics-400 dataset(Carreira & Zisserman, 2017), and further fine-tuned with British Sign Language data(Duarte et al., 2021). As a result, the video representations capture the visual cues and the temporal information. Same as the PHOENIX-2014T, the videos are 25 frames per second, and each frame is passed to the I3D network to get the video representation (I3D features).

5.1.2 BOBSL Data Preparation

In order to implement the BSL translation model training using Transformer network architectures introduced by papers in Section 4.2, the data of the BSL datasets has to be pre-processed, especially the features extraction of sign language video, and rearranged

according to the format mentioned in the previous Section. This section explains the preparation of BOBSL data.

5.1.2.1 Sign Language Video Representations

In the folder “i3d_c2281_16f_m8_-15_4_d0.8_-3_22” under “features”, there are files containing features extracted from each BBC episode at a stride of 4 frames using an I3D model. The I3D features were used for automatic sign spotting using techniques of mouthing cues and dictionary spotting (Albanie et al., 2021). The I3D model was pre-trained with sign recognition on BOBSL data with video pose-distillation, which effectively forces the I3D model to pay attention to the dynamics of the human key points.

The I3D features of the files in this folder serve as video representations or sign language features. Then, they are arranged according to the time range of the subtitle sentence, which will be explained in the next paragraph. For example, file “5172231547180314349.mat” is a matrix of shape (22076, 1024), where 22076 is the number of I3D features. BBC episode ID “5172231547180314349” is an episode of about 59 minutes, thus there are about 88304 frames as the episode video is in 25 fps. Since it is a stride of 4 frames for the features extraction using the I3D model, 22076 features were obtained.

5.1.2.2 Sign Annotations from Automatic Sign Spotting

As mentioned in Section 3.3.2, there are several research working on sign spotting and automatic annotation of the dataset BOBSL and its sunset BSL-1K:

1. “Scaling up sign spotting through sign language dictionaries” (Albanie et al., 2020)
2. “Sign language video retrieval with free-form textual queries” (Duarte et al., 2021)
3. “Automatic dense annotation of large-vocabulary SL videos” (Momeni et al., 2022)

Sign spotting means to determine whether and when a subtitle word is signed. These papers exploit techniques, including mouthing cues spotting, SL dictionary spotting, and attention mechanism, to automatically spot the signs.

In the folder under ‘spottings’ of BOBSL, there are multiple JSON files such as “mouthing_spottings.json”, “dict_spottings.json”, and “attention_spottings.json”. Each JSON file consists of sign words spotted by various techniques mentioned. Nonetheless, there is a “auto_dense_annotations.pkl” file online provided by the paper of Momeni et al.(2022). Compared to the JSON files, the Pickle file is better organized and completed. It contains a collection of all sign instances spotted by all previous sign-spotting efforts using different techniques. The Pickle file is a dictionary with five keys: 'episode_name', 'annot_word', 'annot_time', 'annot_prob', and 'annot_type'. Each key corresponds to a list of values that store the information of a sign annotation. The file contains 5901131 sign instance annotations and 89870 different sign words. Figure 15 shows the dictionary structure of the sign spotting annotations Pickle file. As shown in Figure 15, a sign instance meaning ‘team’ was signed in video “5492816924754876821.mp4” at 427.6ms since the beginning of the video, with the spotting probability of 0.99 using the Mouthing Cues Spotting(Albanie et al., 2020) technique.

```
{
 'episode_name': ['5492816924754876821.mp4', '5129558040617440365.mp4',
'5396381165076599176.mp4', '5921890600084285785.mp4', ...],
 'annot_word': ['team', 'create', 'die', 'neck', ...],
 'annot_time': [427.6, 2607.84, 1017.12, 1984.44, ...],
 'annot_prob': [0.99, 0.89, 0.75, 0.85, ...],
 'annot_type': ['M*', 'D*', 'P', 'E', ...]
}
```

Figure 15 - The dictionary structure with four sign instance annotations as examples

According to Momeni et al.(2022), one of the main goals of automatic sign spotting is to significantly increase the supply of datasets with large-scale vocabularies for continuous sign language recognition, which aims at recognizing glosses from sign videos. Therefore, the sign instances automatically spotted or recognized actually have a similar role to that

of the glosses which are annotated manually. There are two major usages for these spotted sign instances information provided, including being used for (i) gloss supervision of the Sign2(Gloss+Text) training approach and (ii) sentence-level data filtering.

Before using those sign instances spotted as glosses, they are extracted from the dictionary file, shown in Figure 15, according to the time range of the subtitle sentences. Figure 16 shows an example of the sign instances being extracted and arranged as glosses representing the sign coniology and the meaning of the subtitle sentence.


```
"5085344787448740525_496": {
  "start": "00:33:08.847",
  "end": "00:33:24.719",
  "gloss": "FOUR MONTH PLAN UNDRESSED BETTER BUILD BUILDER BUILDING ENSURE MAKE OPEN READ SET TELL GOOD",
  "text": "We'll be launching within the next three to four months off-plan, certainly nothing to show
  for, because the building is very much as you have seen in its undressed state at the moment and
  getting worse before it'll get better."
},
```


Figure 16 - Example of sign instances extracted as glosses representing the meaning of the subtitle, though there are sometimes many sign words that are derivatives and do not appear in the subtitle sentence

Gloss supervision of the Sign2(Gloss+Text) training

Regarding the need for glosses of SLT model training with S2(G+T)-Trans, sign instance annotations by automatic sign spotting are used as the gloss annotations in the experiment of this paper. The gloss information is only useful for training the SLT model using S2(G+T)-Trans, other models only require sentence annotations.

Nonetheless, not all spotted sign instances provided by the automatic annotation are used. Within the time period of a subtitle sentence, the sign instances usually contain multiple derivatives of a stem word and invalid words, as shown with some examples in Figure 16. Also, some sign instances do not appear in the corresponding subtitle sentence. According to the gloss and sentence pair of the original PHOENIX-2014t dataset used for those training, there are mostly stem words without multiple

derivatives and invalid words, and all glosses exist in the spoken language sentence.

Therefore, some spotted sign instances are filtered out. First, sign instances, spotted by the automatic annotation, that are within the time period of the subtitle sentence are checked whether they are valid written words using the word corpus of the Python library NLTK, and then stem comparisons are conducted to remove any duplicate derivatives, as shown in Figure 17.

```
if nltk:
    sign_appeared.append(sign.upper()) if sign in set(words.words()) else None
```

```
stemmer = PorterStemmer()
text_stems = [stemmer.stem(token) for token in word_tokenize(text_str)]
for word in gloss_words:
    if stemmer.stem(word) in text_stems:
        appeared_glosses.append(word)
```

Figure 17 - Python scripts related to sign instances filtering

5.1.2.3 Sentence Filtering using Auto-Spotted Signs

Since the signing in BOBSL is an interpretation of the BBC episode content, it is assumed that the subtitle sentences may usually be different from the direct translation of the signing. The interpreter may sign according to the context and meaning of the episode content. As a result, there are limitations to the extent that sentence-level alignment could help in making high-quality sign language and spoken language sentence pairs, let alone the manual alignment error. Figure 18 shows an example that sign-aligned subtitles consist of possibly unaligned sign instances within them, where the glosses are automatically spotted.

```
"5391892065246550398_796": {
    "start": "00:51:26.832",
    "end": "00:51:30.041",
    "gloss": "OLDER",
    "text": "I believe my opinion is just as valid as anybody older or younger than me."
},
"5391892065246550398_797": {
    "start": "00:51:28.244",
    "end": "00:51:36.081",
    "gloss": "ANIMAL OLDER BECAUSE MY WHY",
    "text": "My skills and my love lie in actually working with animals hands-on."
},
"5391892065246550398_798": {
    "start": "00:51:36.320",
    "end": "00:51:37.983",
    "gloss": "SENSUAL",
    "text": "Why should you win this?"
```


Figure 18 - Automatic spotted signs indicate unaligned sign instances among several consecutive subtitle sentences

Therefore, it is assumed that the greater the number of sign instances that (i) are spotted with high probability within the time range of a sign-aligned subtitle, and (ii) their sign words appear in the sentence subtitle, the more closely the subtitle sentence resembles a direct translation of the signing. Various sentence filters were devised based on this assumption, including the probability of automatic sign spotting, whether there is any sign spotted, and the minimum count of spotted signs appearance. Table 8 shows various numbers of sentence data resulting from different filtering parameters applied.

Signs Spotting Filters			Sentences
Probability	Min. Appearance	Spotted Sign>0	
-	-	False	31479
-	-	True	23835
0.5	1	True	14633
0.9	1	True	9598
0.5	2	True	6539
0.7	2	True	5687
0.9	2	True	2895

Table 8 – Sentences amount corresponding to the filtering based on the auto-spotted signs

5.1.2.4 Data Rearrangement

Since sentence-level sign language data is required, data is rearranged according to the sign-aligned subtitle sentences. Python scripts were written for looping through subtitle sentences of each episode's sign-aligned subtitle ".vtt" file. For each sentence, the starting time and ending time are extracted to obtain the time range in milliseconds.

After that, sign language video features and sign instance annotations are retrieved according to the time range.

```
00:39:23.919 --> 00:39:30.714
Rejoice national fish herring from special dancing.
```

Figure 19 - A sign-aligned subtitle sentence with a corresponding time range in a ".vtt" file

```
{'name': '5172231547180314349_276', 'signer': 'Signer01', 'gloss': 'fish herring dancing national', 'text': 'Rejoice nat
ional fish herring from special dancing.', 'sign': tensor([[0.3386, 0.0000, 0.0000,  ..., 0.0000, 0.0000, 0.1744],
        [0.1617, 0.0038, 0.0000,  ..., 0.0035, 0.0000, 0.0190],
        [0.0030, 0.0010, 0.0000,  ..., 0.2137, 0.0000, 0.1200],
        ...,
        [0.0005, 0.0507, 0.3347,  ..., 0.0000, 0.3569, 0.0000],
        [0.0119, 0.0701, 0.1877,  ..., 0.0000, 0.5303, 0.0000],
        [0.0187, 0.0413, 0.1586,  ..., 0.0000, 0.1995, 0.0000]])}
```

Figure 20 - Arranged sign language information for a sentence-level data entry

Figure 19 and Figure 20 show an example of arranging information of a sign-aligned subtitle sentence to the format of data used by training with the S2(G+T)-Trans, S2T-LCU-Trans, S2T-GA-Trans approaches, which shows how: (i)sign language video features are arranged for the value of ‘sign’, (ii)sign instance annotations are assigned for the value of ‘gloss’, (iii)subtitle sentence be the value of ‘text’. In the example, the value of key ‘name’ indicates that it is the 276th subtitle sentence of BBC episode ID “5172231547180314349”. The value of the key ‘sign’ is a 2D tensor with the shape 43*1024, which means there are 43 I3D features representing the sign language video visual features or sign features. As the fps of the sign language video is 25, there are 170 frames for 6.795 seconds (39:23.010-39:30.714) time range of the subtitle sentence. Thus, 43 I3D features were retrieved considering I3D model extraction using a stride of 4 frames. Besides,

5.1.3 BSLCP Data Preparation

The data rearrangement for training the SLT model on BSLCP is similar to that of the BOBSL. BSLCP does not provide extracted sign language video features like since it is not a machine-learning dataset. Therefore, feature extraction was conducted for BSLCP, which is explained in the next Section.

5.1.4 Sign Language Video Features Extraction

Feature extractions were performed for extracting sign language video features, or say visual representations, of BOBSL and BSLCP. The feature extraction models used include 2D CNN and I3D models.

5.1.4.1 Visual Features Extraction using 2D CNN Model

Only BOBSL, as a machine learning dataset, has provided the I3D features extracted from each episode at a stride of 4 frames using an I3D model, while BSLCP does not. Despite that, in addition to training the SLT model using the I3D features provided by BOBSL, we also attempted to train the model with features extracted using a method similar to that used by S2(G+T)-Trans (Section 4.2.1), which were obtained, that is frame-level extraction using a 2D CNN model.

As mentioned in Section 5.1.1.1, the 2D CNN is an Inception network with GoogleNet architecture (Szegedy et al., 2014) and its model was fine-tuned for sign language recognition in a CNN+LSTM+HMM setup (Koller et al., 2020). However, Koller et al.(2020) did not publish the weights of the fine-tuned 2D CNN model. Due to the limitation of time, the CNN+LSTM+HMM setup was not built for training and obtaining the fine-tuned weights. Instead, only the weight of GoogleNet is used as a 2D GoogleNet model to perform the frame-level feature extractions.

5.1.4.2 Visual Features Extraction using I3D Model

Although the S2T-GA-Trans approach experiment by A. Yin et al.(2023), mentioned in Section 4.2.3, uses the PHOENIX-2014-t dataset, they did not use the 2D frame-level features or representations provided by Camgoz et al. (2020). Instead, they used the I3D visual features extracted from the TSPNet project(Li et al., 2020), which used I3D model weights pre-trained for the task of isolated sign recognition on the WLASL dataset and feature extraction code provided by the WLASL project (Li et al., 2020), in a sliding window of 8 and a stride of 2.

In this paper, we also used the features extraction code provided by the WLASL project (Li et al., 2020) for loading the pre-trained weights of the I3D model and extracting the frame-level visual representations or features. Since we are using the BSL dataset, we used the weights provided by Varol et al.(2021) that are from a further fine-tuned I3D model using 700K sparse automatic sign annotations in BOBSL based on the pre-trained weights for the isolated SLR task on WLASL dataset.

5.1.5 Sentence Cosine Similarity

Sentence cosine similarity between the spoken language sentences was calculated for data used to train the SLT model with the S2T-GA-Trans approach mentioned in Section 4.2.3. A. Yin et al.(2023) used cosine similarity between the spoken language sentences to improve the accuracy of the SLT model by transferring knowledge from the spoken language model learned on a rich corpus of data.

As suggested by A. Yin et al.(2023), we use the “distiluse-base-multilingual-cased-v1” model from the Sentence-Transformers project(Reimers & Gurevych, 2019) to encode the spoken language sentences and calculate the similarity among them. Figure 21 shows a brief code of a Python function calculating cosine similarity among a list of sentences. For instance, if there are 18179 spoken language sentences, the cosine similarity result is represented as a tensor with shape (18179*18179), as shown in Figure 22.

```python
from sentence_transformers import SentenceTransformer, util
def calculate_cos_sim(sentences):
    model = SentenceTransformer('distiluse-base-multilingual-cased-v1')
    sentences_embeddings = model.encode(sentences)
    cos_sim = util.cos_sim(sentences_embeddings, sentences_embeddings)
    return cos_sim
```

Figure 21 - Brief Python function for cosine similarity calculation

```
tensor([[ 1.0000, -0.0328,  0.0057,  ...,  0.1516,  0.0453,  0.0033],
        [-0.0328,  1.0000,  0.1342,  ...,  0.1863, -0.1318,  0.1468],
        [ 0.0057,  0.1342,  1.0000,  ...,  0.1661,  0.1243,  0.1478],
        ...,
        [ 0.1516,  0.1863,  0.1661,  ...,  1.0000,  0.0847,  0.0951],
        [ 0.0453, -0.1318,  0.1243,  ...,  0.0847,  1.0000,  0.1418],
        [ 0.0033,  0.1468,  0.1478,  ...,  0.0951,  0.1418,  1.0000]])
torch.Size([18179, 18179])
```

Figure 22 - Result example for cosine similarity of 18179 spoken language sentences

5.2 Summary of SLT Model Training

5.2.1 Neural Architecture

Model Architecture	Encoder Layers	Decoder Layers	Attention Heads	FFN Dim.
S2(G+T)-Trans	3	3	8	512
S2T-LCU-Trans	3	3	8	512
S2T-GA-Trans	2	2	8	512
S2T-Trans	6	3	4	1024

Table 9 - Summary of the model training hyperparameters

5.2.2 Training Summary

Model	Dataset	Sentences	Words	Video	Visual Representations
S2(G+T)-Trans	BOBSL	31479	337248	135732s	I3D features by BOBSL
S2T-LCU-Trans	BOBSL	31479	337248	135732s	
		23835	295453	124178s	
		14633	191958	83408s	
		9598	130939	57567s	
		6539	104933	46047s	
		6539	104933	46047s	Non-pre-trained 2D CNN
		5687	92272	158777s	I3D features by BOBSL
		2895	48950	21690s	
	BSLCP	28853	207093	66852s	fine-tuned I3D model
S2T-GA-Trans	BOBSL	31479	337248	135732s	I3D features by BOBSL
	BSLCP	18179	203551	58359s	fine-tuned I3D model
S2T-Trans	BOBSL	31479	337248	135732s	I3D features by BOBSL

Table 10 - Summary of all the SLT model training experiments

6 Results

This section shows the experiment results of the SLT model trained on the two BSL datasets with the four training approaches.

6.1 Results of Training on BSL Datasets

Table 11 shows the BLEU results of training the SLT models with the Transformer architectures mentioned on BSL Datasets, namely BOBSL and BSLCP. Generally, the SLT models trained on BSLCP achieve higher accuracy than that of BOBSL. For the Transformer architecture, the SLT model trained using the Sign2Text architecture proposed by Paper 4 possibly achieves the best transformation result. However, it was failed to train it on BSLCP.

Transformer Architecture	Dataset	Sentences & Words	Video Length	BLEU			
				1	2	3	4
S2(G+T)-Trans	BOBSL	31479	135732s	8.17	2.62	0.80	0.24
S2T-LCU-Trans	BOBSL	31479	135732s	8.57	2.81	1.07	0.43
	BSLCP	28853	66852s	12.35	4.13	1.87	0.79
S2T-GA-Trans	BOBSL	31479	135732s	5.21	1.69	0.58	0.22
	BSLCP	18179	58359s	9.65	2.42	0.76	0.35
S2T-Trans	BOBSL	31479	135732s	13.5	4.06	1.83	0.68

Table 11 – BLEU scores of SLT models trained on different datasets and approaches

6.2 Results of Training on BOBSL with Filters

Considering that BOBSL is a scalable broadcast interpretation-based large-scale dataset, multiple training attempts were conducted to experiment with its potential usage in training SLT models with high practicability in the future. Especially, the sign instances spotted by automatic annotation models, mentioned in Section 5.1.2.2, are utilized for filtering sentence pairs. Each spotted sign instance is provided along with its corresponding probability, which is between 0 and 1, and the appeared time in the video.

Based on the probability, SL sentence filtering was conducted with various metrics, including (i) the spotting probability, (ii) the amount of the spotted sign instance that appeared in the

subtitle sentence, and (iii) whether a sentence without a sign spotted within the sentence time range should be filtered out.

The training attempts using different filter configs were only performed using the Sign2Text approach of S2T-LCU-Trans The results are shown in Table 12.

Signs Spotting Filters			Sentences	BLEU			
Probability	Min. Appearance	Spotted Sign>0		1	2	3	4
-	-	False	31479	8.57	2.81	1.07	0.43
-	-	True	23835	10.12	3.31	1.19	0.45
0.5	1	True	14633	11.87	4.08	1.45	0.60
0.9	1	True	9598	11.40	3.83	1.50	0.74
0.5	2	True	6539	12.26	4.13	1.66	0.77
0.7	2	True	5687	11.14	3.98	1.67	0.73
0.9	2	True	2895	10.10	3.48	1.18	0.46

Table 12 - BLEU scores of SLT models trained on BOBSL with sign-spotted filters

Generally, BOBSL data with stricter filtering applied yields better BLEU scores. However, the more filtering applied, the less number of sentences available for training.

6.3 Results of Training with Different Features

SLT models trained with video representations or sign language features extracted using fine-tuned I3D models generally achieve better results than the counterparts of using 2D CNN, which is an Inception network with GoogleNet architecture, without fine-tuning sign language data. Training results of the SLT model using the S2T-LCU-Trans approach trained on BOBSL with different visual representations of sign language videos are shown in Table 13.

Sentences	Features Extraction Model	BLEU			
		1	2	3	4
6539	Fine-tuned I3D	12.26	4.13	1.66	0.77
6539	Non-pre-trained 2D CNN	10.21	2.68	0.70	0.23

Table 13 - BLEU scores of SLT models trained on BOBSL with features extracted by different model

7 Discussion

Based on the training results, this section evaluates the SL datasets originally or currently used, the utilization of large-scale auto-annotated datasets, data pre-processing, and the four SLT model training architectures.

7.1 Comparison of Datasets

In this section, the usability and characteristics of datasets for SLT tasks including PHOENIX2014-t, BOBSL, and BSLCP will be discussed based on the translation performance of the SLT models trained.

7.1.1 PHOENIX2014-t and BSL Datasets

The best BLEU scores SLT models trained on BSL datasets achieved were far lower than that of PHOENIX2014-t. The possible reasons may be the scale and variety of the dataset.

PHOENIX is the most standardized dataset available, which is the most used dataset for benchmarking SLT models (Núñez-Marcos et al., 2023). However, the dataset is limited in various aspects including (i) a small number of signers and vocabulary, (ii) limited in linguistic variety and specific domains like weather forecasts and emergency situations, and (iii) lack of visual variability(Shi et al., 2022).

When it comes to BSLCP, it covers more domains and sign words than PHOENIX2014-t. However, as Camgoz et al.(2021b) mentioned, BSLCP was not collected with vision-based sign language research and lacks the necessary duplication and inter-/intra-signer variance on shared content. A low ratio between video hours per unique sign word significantly decreases the SLT model performance.

BOBSL covers a significantly broader domain of discourse and more open vocabularies, but the results are similarly bad or worse than that of BSLCP if no subtitle sentence filters are

applied. The same situation occurred for other large-scale datasets, such as DGS Corpus (Zhang et al., 2023), and the DSGS broadcast dataset(Müller et al., n.d.). Besides, the supervision of the signed content is limited, making BOBSL relatively weaker and more noisy than PHOENIX2014-t, which will be further discussed in the next section.

Compared to the BSL datasets including BOBSL and BSLCP, sign words of PHOENIX2014-t are significantly less but have greater repeating occurrence. The SLT models trained on PHOENIX2014-t are generally easier to achieve higher BLEU-4 scores, usually over 15. Despite that, they are significant over-fitting translation models, which means they also have low performance in domains other than weather forecasts and emergency situations. Tarrés et al.(2022) also obtained low BLEU scores for the SLT model trained on another large-scale dataset, similar to BOBSL. They refer to the reason that the dataset has a much lower ratio between video hours and vocabulary size compared to that of the PHOENIX2014-t, which indicates that it is a relatively much more complicated dataset with an insufficient amount of data.

7.1.2 BOBSL and BSLCP

Without filtering the sign sentences for training, SLT models trained on BSLCP generally achieve better than those trained on BOBSL. As Momeni et al.(2022) mentioned, the creation of the BOBSL dataset included limited supervision of the signed content and annotation, which makes the dataset relatively weak and noisy. They also found that the presence of a word in the subtitle sentence does not necessarily imply that the word is signed in the video or signed in totally different ways.

One major reason that contributes to the BOBSL noisiness is the interpretation nature of the subtitles. Different from BLSCP in which the annotations are directly translated from the native signer, the subtitles in BOBSL are transcripts of the audio in spoken language, while

the sign interpreter does not necessarily sign what the audio said but according to the context and the meaning of the BBC episode content instead.

Moreover, subtitles are originally temporally aligned with the audio content but not necessarily with the signing. In the SLT model training experiment of this paper, only episodes of BOBSL manually aligned were used, introducing possible alignment manual errors.

7.2 Large Scale Interpretation-Based Dataset

SLT models trained on the large-scale interpretation-based dataset BOBSL, without sentence filtering, achieved the worst performance, compared to using datasets BSLCP and PHOENIX2014-T. Low SLT performance was generally obtained as well for the experiments of other research papers using BOBSL. In the paper that introduces the BOBSL dataset, Albanie et al.(2021) actually conducted a baseline SLT model training experiment on BOBSL, which obtained only 1.0 BLEU-4 scores. They mentioned that the Transformer struggles to capture the meaning of the sentences of BOBSL. Various other experiments on BOBSL obtained a similar performance of the SLT model, such as 1 BLEU-4 (Tarrés et al., 2023) and 1.27 BLEU-4 (Sincan et al., 2023).

Despite that, the lack of large-scale datasets in the wild is a central challenge in developing SLT technologies that serve real-world sign language users(Shi et al., 2022). As mentioned in the previous sections, the model trained on the benchmark SL dataset PHOENIX-2014-t has little real-world applicability, since it is restricted to specific domains, limited to small vocabularies, and lacks visual variability. In terms of the manually annotated linguistic corpus BSLCP, despite that it is an extensive ongoing research effort spanning more than a decade, only a relatively small fraction of the SL data is annotated(Schembri et al., 2013). Scaling up BSLCP requires highly skilled annotators and vast labour investment.

Constructing datasets like BOBSL, a scalable broadcast interpretation-based large-scale SL dataset, along with various automatic technologies such as subtitle auto-alignment and sign auto-spotting, is a potential solution to the data scarcity problem of developing SLT technologies. Therefore, the following sub-sections will discuss the attempts to make use of information provided by those automatic technologies to better utilize the large-scale dataset BOBSL.

7.2.1 Training with S2(G+T)-Trans Approach

As mentioned in Section 5.1.2.2, BOBSL provides sign words that were automatically spotted using techniques, including mouthing cues spotting, SL dictionary spotting, and attention mechanism. Those sign instances were employed as glosses in this paper for training the SLT model based on the state-of-the-art Sign2(Gloss+Text) approach proposed by Camgöz et al.(2020), mentioned in Section 4.2.1. In the approach, glosses serve as supervision signals during the training of the SLT model. Since glosses are monotonically aligned to the sign language video, they can provide stronger supervision than sentence text in spoken language, which facilitates the networks to learn more meaningful spatiotemoral representations of signs(Shi et al., 2022).

Therefore, one of our key observations pertains to the feasibility of employing automatically spotted sign instances as gloss supervision during Sign2(Gloss+Text) model training. As revealed in Section 5.2.2, the SLT model trained using the Sign2(Gloss+Text) approach yields inferior performance compared to the S2T-LCU-Trans approach. This discrepancy suggests that the sign instances spotted automatically may not be enough and optimally fulfilling their role within the gloss supervision function during the Sign2(Gloss+Text) approach training.

The amount of glosses, which are served by the sign instances automatically spotted, is far from enough to represent the meaning of the subtitle sentences in general. The average gloss-to-word ratio of BOBSL, which is the average ratio between the number of glosses and

the number of words in the sentence, is far lower than that of the PHOENIX2014-t. Assuming that glosses in the gloss lists of sentences in PHOENIX2014-t can represent the major meaning of most sentences, only less than half of the meaning of BOBSL sentences is conveyed by the glosses on average, considering the gloss-to-text ratio, as shown in Table 14.

Dataset	Spotting Probability	Avg. Gloss-to-Word Ratio
PHOENIX2014-t	-	0.53
BOBSL	0.3	0.23
	0.5	0.18
	0.7	0.16

Table 14 - Gloss-to-Word ratio of PHOENIX2014-t and BOBSL

Moreover, the spotted signs may not be correct, especially for those with spotting probabilities lower than 0.6. Even though the sign instances that do not appear in the subtitle sentences were already filtered out, the appeared sign words may also be incorrect deteriorating the performance of the trained SLT models.

7.2.2 Automatic Spotted Signs for Sentences Filtering

As the results shown in Section 6.2, with more sign spotting filters applied, the performance increases generally but decreases when it comes to the strictest filtering. The amount of trainable sentences significantly decreases as the filtering becomes stricter. Based on the result, Figure 23 shows a linear graph better representing how the filtering might be related to the SLT model performance.

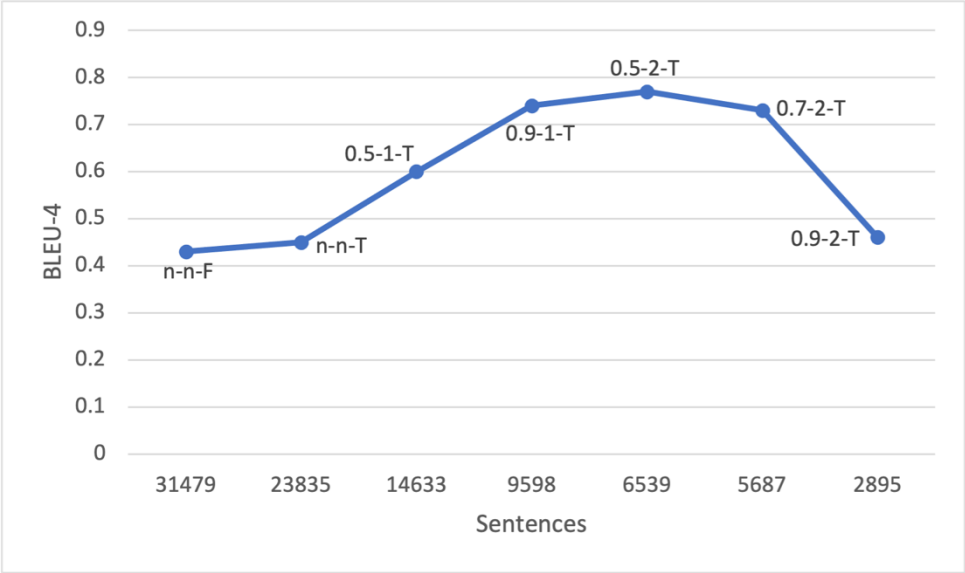

Figure 23 - Linear graph of the relationship between filtered sentences and BLEU-4

The resulting trend might indicate the sentence filtering approaches devised based on the auto-spotted signs, to a certain extent, successfully filtered out inconsistent signing and spoken language pairs, resulting in improved SLT model performance. However, stricter filters result in small amount of sentences, which significantly hinders further improvement. Considering that BOBSL contains sentences from large domains of discourse, the lower the repeating occurrence of the sign words it is, the more difficult the model to capture the meaning, as mentioned in Section 7.1.1.

7.2.3 Model Inference Performance

Sentences	Text
Reference	I emailed to ask whether there had been another deaf person in the world.
Hypothesis	I think that I have to sign for a deaf person and i have been to the same signs.
Reference	will the business survive Saba's first season in charge?
Hypothesis	and this is where the first time to run the first time.
Reference	The Wye Valley is certainly a great place to go walking and canoeing.
Hypothesis	the valley is a bit of a major eruption?

Table 15 - Examples of the inference performed by a SLT model trained on BOBSL

Table 15 shows some inferencing examples of the SLT model trained on BOBSL, in which "Reference" represents the sentence annotations, while "Hypothesis" is the sentence predicted by the SLT model. As we can see, the inference does not perform well, as it struggles to capture the meaning of the sentences of BOBSL. Most of the time, the SLT

models only correctly predicted at most several sign instances within a sentence. Some phrases are observed to be predicted frequently, such as “I think”, “the first time”, “a bit of”, “is the same” etc. Similar to the observation of Shi et al.(2022), the model is learning the most frequent words in the vocabulary, but failing to provide meaningful predictions from the SL videos. The ratio between the training data and the vocabulary size is too low for the model to learn for the inference task.

7.3 Pre-Processing the Training Input

Various pre-processing operations are performed for the input data, including the sign language videos and spoken language text before it is passed into the Transformer network for training. Different pre-processing approach contributes to the performance of the SLT model trained. This section discusses the observation of the importance of the selection of pre-processing operations based on the experiment results obtained in this paper.

7.3.1 Visual/Sign Representation Extraction

As shown in Section 6.3, SLT models trained using visual representations or sign language features provided by BOBSL, which were extracted by a fine-tuned or I3D model, generally perform better than those extracted by a 2D CNN model without pre-training on sign language data. The 2D CNN model was an Inception network in GoogleNet architecture, details in Section 5.1.1.1.

A comparison between the pre-trained 2D CNN Inception model and the pre-trained I3D model was not performed, since the weights of the pre-trained 2D CNN model trained in the CNN+LSTM+HMM setup(Koller et al., 2020) were not obtained. Despite that, the results still, to a certain extent, prove the assertion of Albanie et al.(2020) that I3D models significantly outperform their 2D pose-based counterparts for extracting visual representations. Especially, the I3D model that they used was pre-trained with isolated sign language recognition tasks.

In addition to the CNN architecture difference, it is also assumed that the pre-training process of the I3D model helped it to learn the visual representation better. As Shi et al.(2022) mentioned, pre-training the I3D network on an isolated sign recognition task can provide more direct supervision for the convolutional layers than full sign language translation or continuous sign language recognition tasks. Since the pre-training of the 2D CNN Inception model was a CNN+LSTM+HMM continuous sign language recognition setup, there may not be sufficient direct guidance for learning the visual representation, as observed in prior work(Camgoz et al., 2018).

7.3.2 Spoken Language Text Preprocessing

Among the four papers, only papers on S2T-GA-Trans and S2T-Trans performed text segmentation or tokenization on the spoken language annotations before they were passed to the Transformer network for training. The annotation text is lowercase and segmented by the BPEmb model and Sentencepiece model in S2T-GA-Trans and S2T-Trans respectively. They consider this practice to bring an improvement over the conventional method that treats each word as a unit of the sequence since it leads to an expansive vocabulary and an inability to account for those previously unseen words.

The text tokenization practice on the spoken language text input might have contributed to the better performance of the SLT model trained using the S2T-Trans. In the paper on S2T-Trans, Tarrés et al.(2023) did experiments to compare the performance difference of SLT models trained with or without text preprocessing and different vocabulary sizes. The one performing text preprocessing results in 58% better performance than the one without, while 7000 vocabulary yields the best result.

Nonetheless, there are other possible reasons for a model trained using the S2T-Trans approach to achieve better performance, such as the training hyperparameters, which will be mentioned in the next section.

7.4 Transformer Training Approaches

Based on the results, this section discusses the training implementation of the Transformer architectures, including S2(G+T)-Trans, S2T-LCU-Trans, S2T-GA-Trans, and S2T-Trans.

7.4.1 S2(G+T)-Trans Approach

As mentioned in Section 9.2.1, using the automatic spotted sign instances in BOBSL as glosses, the gloss-to-word ratio to too low that the glosses cannot represent the meaning of the sentence. The glosses are not enough to serve as supervision signals to the SLT model training, which may even negatively affect the learning of the model.

7.4.2 S2T-Trans Approach

As the results shown in Section 6.1, with the BOBSL data without any filtering, the SLT model trained using the S2T-Trans approach achieves the highest BLEU scores. One of the possible reasons is the hyperparameters of the network. Table 16 shows the details of the network training hyperparameters of the 4 papers.

Model Architecture	Encoder Layers	Decoder Layers	Attention Heads	FFN Dimension	BLEU			
					1	2	3	4
S2(G+T)-Trans	3	3	8	512	8.57	2.81	1.07	0.43
S2T-LCU-Trans	3	3	8	512	8.57	2.81	1.07	0.43
S2T-GA-Trans	2	2	8	512	5.21	1.69	0.58	0.22
S2T-Trans	6	3	4	1024	13.5	4.06	1.83	0.68

Table 16 – Transformer Hyperparameters and results of the four models trained on BOBSL

In general, it is observed that increasing the number of Transformer network parameters improves the performance of the model trained. However, in terms of low-resource languages with limited annotated data or linguistic resources, increasing the number of model parameters hinders performance(K. Yin & Read, 2020).

In S2T-Trans, Tarrés et al.(2023) studied the effect of using a deeper and shallower Transformer. They attempted model training with different numbers of layers in the encoder and decoder, attention heads, feed-forward layer dimension, and embedding

dimensions. Their results show that the smaller the models are, the better the model performs. According to their loss curves, there was a substantial amount of overfitting for the large models, since tuning that many parameters require a large amount of data, which is impossible given the current data scarcity of sign language datasets. Camgöz et al.(2020), who proposed the S2(G+T)-Trans, also observed an overfitting phenomenon in the experiments of training the SLT model using different number of layers of the symmetric Transformer. They found that the 3 layers of encoder and decoder yield the best performance, while a greater number of layers results in overfitting.

Considering the overfitting, Tarrés et al.(2023) added regulation by adding dropout, label smoothing, and weight decay to make the model more robust to overfitting. Their results show that the large model paired with regularization techniques outperforms the smaller models. The best-performing model, which is an asymmetric model with 6 layers in the encoder and 3 layers in the decoder, is shown in Table 17 with its set of hyperparameters and regulations.

Encoder Layers	Decoder Layers	Attention Heads	FFN Dimension	Dropout	Weight Decay	Label Smoothing
6	3	4	1024	0.3	0.1	0.1

Table 17 - Hyperparameters of the best performing model of S2T-Trans approach

Since the learning rate is dependent on the number of model parameters, Tarrés et al.(2023) tuned it along with the hyperparameters related to architecture size. They introduced LR scheduling of cosine with warm restarts, which has been proven to perform better than alternatives(Loshchilov & Hutter, 2016).

In short, one of the possible reasons that the SLT model trained using Paper 4's approach outperforms the others is that it is a relatively larger asymmetric model with more parameters paired with regularization techniques and an optimal learning rate during the training process.

7.4.3 S2T-LCU-Trans Approach

Compared to S2(G+T)-Trans, the SLT model trained using S2T-LCU-Trans has higher accuracy. As mentioned in Section 7.2.1, with low gloss-to-sentence word ratio or density, glosses may interfere instead of supervising the learning of the SLT model using the S2(G+T) method. Without explicitly considering the gloss information, the SLT model trained with the S2T-LCU-Trans method performs better.

In the experiment of Voskou et al.(2021), the SLT model trained using the S2T-LCU-Trans approach performed better than the model trained using S2(G+T)-Trans using the PHOENIX-2014T dataset, in which the gloss information is sufficient. Thus Voskou et al.(2021) claim that the S2(G+T) method is a gloss-free approach capable of training an SLT model that performs even better than the S2(G+T)-Trans method which requires gloss annotations. However, A. Yin et al.(2023) mentioned that the S2T-LCU-Trans was not considered the gloss-free method as the gloss information is already implicitly included in the extracted visual features. Lin et al.(2023) also considered the S2T-LCU-Trans is actually a gloss-based approach.

Voskou et al.(2021) mentioned that the use of the Stochastic Transformer with LCUs and Stochastic WTA activations improves the model's ability to capture complex patterns in the data. Therefore, it is implied that the S2T-LCU-Trans approach helps the SLT model train better with gloss information provided, which is implicit in the visual features, but not necessarily better when it comes to training on a completely gloss-free dataset. SLT model trained using the S2-LCU-Trans on a dataset without any gloss information may still perform worse than that trained with gloss information.

In terms of data totally without gloss information, according to the experiment results of this dissertation using the BOBSL dataset without gloss annotations, a deeper network using conventional Transformer architecture paired with regulation techniques, which is the

S2T-Trans mentioned in the previous section, may still perform better than a Stochastic Transformer with LCUs and a shallower network.

In addition to the performance of the SLT model trained, with their trained network compressions scheme, the memory requirements of the model at inference time are reduced. This would be helpful in terms of bringing the SLT model into real-life inferencing applications. SLT models trained with a deeper Transformer network may have higher accuracy, but their corresponding higher memory requirements make it costly and less practical for real-life SLT inference applications.

7.4.4 S2T-GA-Trans Approach

Before the experiments were conducted, it was theoretically predicted that the accuracy of the SLT model trained with the S2T-GA-Trans approach would be the highest compared to the others. The main reason is that, among the four replicated SLT model training approaches, S2T-GA-Trans is the only paper that is aware of using SL data without any gloss information and proposes methods to replace the necessity of gloss signal supervision. S2T-LCU-Trans modifies the Transformer using the data with implicit gloss information, thus it is still a gloss-supervised training approach. The S2T-Trans approach uses a conventional Transformer without any modification.

However, the SLT model trained using the S2T-GA-Trans approach achieves the worst accuracy in the experiment of this paper. There are possible reasons related to the implementation of sub-word segmentation using the BPEmb model for text embedding, and the calculation of cosine similarity among sentences for providing global information to the training. Besides, as mentioned in Section 7.4.2, the fact that this approach uses the shallowest Transformer network, which is 2 encoders and 2 decoders, may also contributes to its relatively less accurate performance.

For the sub-word segmentation model BPEmb, as mentioned in Section 4.2.3, it is used for text segmenting and initialization of the text embedding layer. In the experiment, the word embedding file and model file, which are “en.wiki.bpe.vs10000.d300.w2v” and “en.wiki.bpe.vs10000.model” respectively are used. The segmentation results using the English BPE model on the BOBSL dataset might be less favorable for SLT model training than that of the original Germany BPE segmentation on PHOENIX-2014t. Also, since Núñez-Marcos et al.(2023) have mentioned that BLEU computation is highly dependent on factors especially the tokenization used, it is possible that the BPEmb model for English significantly affects the BLEU scores.

In terms of the cosine similarity among sentences that help the model capture the global information, its calculation was conducted by transferring knowledge from the spoken language model learned on a rich corpus of data, as mentioned in Section 4.2.3. Therefore, the performance and compatibility of the spoken language model, which is a sentence Transformer model “distiluse-base-multilingual-cased-v1”, is important. The spoken language model used may be less suitable for English or the BSLCP and BOBSL datasets.

Moreover, the cosine similarity calculation on BSLCP and BOBSL may not help or even hinder the model from recognizing patterns and relationships between the signs and gestures in videos. It might work well on PHOENIX-2014t, but not on BOBSL which covers significantly larger domains of discourse. The negative effect may be compounded by BOBSL’s interpretation nature. For instance, two textually similar spoken language subtitle sentences on BOBSL may be more different in terms of the interpretation signing, considering the two sentences appear in possibly more different contexts or BBC-episode topics.

7.4.5 Summary of the Training Approaches

Given that gloss annotations are enough to function as supervision, the SLT model trained with the S2(G+T)-Trans approach should achieve the best accuracy. However, the BSL datasets used in this dissertation do not have gloss annotations. Using signs auto-spotted in BOBSL as glosses are also observed to be unpractical according to the experiment results in this dissertation, as the gloss-to-word density is too low.

Using the conventional Transformer with the S2T-Trans approach, the SLT model trained with a larger network with suitable regularization implemented may result in higher inference performance, as observed in the training results. Nonetheless, the modified Transformer approaches S2T-LCU-Trans and S2T-GA-Trans with methods proposed to help the model learn better without gloss supervision, such as stochastic weight and gloss attention respectively, did not achieve relatively good results in this dissertation.

The S2T-LCU-Trans approach was claimed to achieve higher accuracy than the S2(G+T)-Trans approach even without using any gloss information. However, other papers have proved that it is still a gloss-based/gloss-supervised approach as it experimented on input features with gloss information implicit. In the experiments of this dissertation without any gloss information, the modifications to the Transformer architecture in the S2T-LCU-Trans approach are not observed to bring significant improvement, especially when it is compared to the S2T-Trans conventional Transformer with more layers and regularizations.

The paper on S2T-GA-Trans investigated the effects of the gloss signal, which consists of maintaining attention focus on the temporal gloss-level semantics, and global information that helps the model to recognize patterns and relationships between the signs and gestures in videos. Regarding the effects of gloss signal, they devised to transfer knowledge from the spoken language model for global information. This dissertation suspects that this method may not be suitable for the BOBSL due to its significantly larger domains of

discourse. Also, the performance and compatibility of the text tokenization BPEme model and the spoken language model may affect the performance of the trained SLT model.

8 Limitations

This section evaluates the limitations of the experiments conducted.

8.1 BSL Datasets

For the BOBSL dataset, as mentioned in Section 4.1.1, only the BBC episodes with sign-aligned subtitles can be used for training. Although the dataset has an approximate duration of 1,400 hours and 1.2M sentences, in this paper, only the manually sign-aligned BBC episodes can be used, which are about 37.7 hours and 31479 sentences. Momeni et al.(2022) also worked on the automatic alignment of subtitles and signing, however, the auto-aligned subtitles were not accessible on the BOBSL dataset website or maybe they were not published. Thus, for the BOBSL dataset which covers a large domain of discourse, only an extremely small amount of its data was sign-aligned and available to be used.

For the BSLCP, XX has mentioned that there are gloss annotations for some of the sign videos. However, there was no corresponding gloss annotation found for those sign videos with EAF sentence-level annotations, which are in the “Interview” and “Narrative” categories on the library’s BSLCP website. As a result, this paper did not perform the training of the SLT model using the S2(G+T)-Trans method on BSLCP, which could be a better comparison with the BOBSL dataset in terms of the effect of the gloss density on the SLT model performance trained with S2(G+T)-Trans approach.

In addition to the used BOBSL and BSLCP datasets, there is another BSL dataset that is the European Cultural Heritage Online (ECHO) multilingual corpus for BSL. It aims at providing comparative information about the sign languages of the EU. This dataset's data was not accessed and used for training in the experiments of this paper, which may be a more thorough comparison of the BSL datasets.

8.2 Sign Language Visual Modality

This section talks about the limitations related to the selection and the usage of visual modality in the experiments of this paper.

8.2.1 Pre-trained Weights of 2D CNN Modality

As mentioned in Section 5.1.1.1, the weights of the 2D CNN model pre-trained/fine-tuned for sign language recognition in the CNN+LSTM+HMM setup (Koller et al., 2020) were not published and accessible. Therefore, the current experiment only shows that the 2D CNN, an Inception network with GoogleNet architecture, model without any fine-tuned for visual features extraction is worse than that using an I3D model fine-tuned on sign language recognition tasks. Besides, training with visual features extracted using the I3D model without being fine-tuned on SLR tasks also was not conducted.

It is assumed that if the weights of the pre-trained 2D CNN were obtained, there would be some better comparisons and findings including: (i) a comparison between non-pre-trained and pre-trained 2D Inception or I3D model, which yields a more specific finding on the importance on fine-tuning of CNN, and (ii) a comparison between 2D CNN and I3D models with both pre-trained through sign language recognition tasks, which facilitates an evaluation of how I3D model is better than 2D Inception CNN model.

8.2.2 Fine-Tunning Weights for I3D Modality

In the SLT model training experiments of this paper, using visual features extracted by a fine-tuned I3D model is the selected and best-performing visual modality. As mentioned in Section 4.2.4, the I3D model was fine-tuned through isolated SLR on sign annotations in WLASL and BOBSL. However, Shi et al. (2022) claim that there are several potential problems for the SLT model trained using the feature extraction model pre-trained through isolated SLR tasks.

First, the coarticulation in the continuous signing stream used in translation is not represented in the isolated SL datasets that are used for fine-tuning the extraction CNN. The domain of the isolated signs is significantly mismatched with the continuous signing data. Second, the instances of fingerspelling are far from enough, as the isolated sign data for I3D model fine-tuning mainly consists of lexical signs. The extraction model may struggle to encode the features related to finger-spelling.

Shi et al.(2022) have proposed some mitigation regarding the mentioned problem of fine-tuning the I3D model with isolated SLR tasks. For instance, they proposed to use a lexical sign recognizer and a fingerspelling recognizer. Nonetheless, fine-tuning was not performed in this paper, not to mention applying those recognizers. Thus, this paper did not examine the approaches for better fine-tuning the I3D model for visual feature extraction.

8.2.3 Pose Representations as Visual Modality

In addition to using visual features extracted by the CNN model, including GoogleNet and I3D model, the training code of the S2T-Trans includes the alternative visual input which is the usage of pose representations for the training of the Transformer network. The pose representations can be in the format of poses either obtained by pose estimators MediaPipe or OpenPose. The paper on the S2T-Trans approach initially explored the use of MediaPipe poses as the visual input for the training and obtained 0.8 BLEU-4 scores, 0.2 lower than that of using I3D as a visual feature, which was deemed as an unsatisfactory result and not included as a baseline for the How2Sign ASL dataset (Tarrés et al., 2023).

Despite that, considering that BSL and ASL are two different sign languages, there would be meaningful findings if pose estimators, MediaPipe or OpenPose, were used to obtain the pose representations as input of SLT model training on BSL datasets. For instance, we could experiment with whether pose representation works better than I3D visual features for BSL in training the SLT model, which is opposite from the case of the ASL dataset How2Sign.

8.3 Text Tokenization

The effect of different text segmentation or tokenization models was also not investigated. In this dissertation, S2T-GA-Trans and S2T-Trans use tokenization models BPEmb and Sentencepiece respectively, but they cannot be directly compared since the corresponding SLT models are also trained in different Transformer networks. Due to the limitation of time and personal capability, an examination of the importance of the tokenization of spoken language annotation, ablation text, and comparison of using different models under the same training setting was not performed.

8.4 SLT Models Training

This section focuses on the limitations mainly related to various SLT model training that were not performed especially for those that cannot be implemented without the modification of the training code provided by the papers.

8.4.1 Discussion on Training Approaches

Training of SLT models for BSL was attempted on four different approaches proposed by four papers. Originally, it was attempted to examine and discuss the Transformer architecture of each approach in detail, including:

(i) how each of the winner-takes-all layers, evidence lower-bound maximization, and stochastic weights in S2T-LCU-Trans helps it to perform better

(ii) are the text tokenization, gloss attention, and global information working in replacing the self-attention and the requirement of gloss supervision signal

(iii) how each of the regularization techniques including dropout, weight decay, and label smoothing help the deep asymmetric Transformer network learn better

However, discussion with this level of detail was unable to be achieved in this dissertation due to limited time and my capability. Only a little modification was conducted to the

training code of those papers, and this paper did not perform any training for ablation tests to separate out each component of the Transformer architecture to experiment with their effect and efficiency. There are only general observations discovering possible reasons for each paper's approach based on their Transformer modifications.

8.4.2 Single-Model Transformer

The training approaches attempted in this dissertation were limited to those with publicly available replication codes, in which the Transformer architectures they used are single-model architectures. Single-model architecture means it takes only one type of input data, which is the visual features extracted using CNN for the architectures reproduced in this dissertation. Some of the other recently proposed training approaches with multi-model Transformer architecture proposed were not attempted in this dissertation.

In this dissertation, it is observed that the sign interpretation of BOBSL is heavily reliance on context, which makes the sentence-level data pair less optimal. Sincan et al. (2023) mentioned the problem related to context interpretation as well, given one of the main reasons that the vocabulary size of the SL is significantly smaller than their spoken language equivalent. Regarding the interpretation problem, they proposed a multi-model Transformer that treats the SLT task in a more context-aware manner by three different encoders.

Besides, to train the model better under low-resource conditions, Shi et al. (2022) proposed a multi-model Transformer that guides the SLT model to learn the local discriminative features or visual cues better, which consists of two additional encoders for the local visual modalities.

However, there is no publicly available training replication code found online. Some detailed information on the multi-model Transformer architectures mentioned is provided in the Appendix.

8.5 Evaluation Metric

In this paper, BLEU was used as the evaluation metric of the experiment in training different SLT models. The limitations regarding the evaluation metric are related to the calculation approach for BLEU scores and other possible evaluation metrics that are not used in this paper.

8.5.1 Calculation of BLEU Scores

BLEU(Papineni et al., 2002) is used as the evaluation metrics of the SLT model trained using the four papers. Among the four papers, papers of S2(G+T)-Trans, S2T-LCU-Trans, and S2T-GA-Trans do not mention any information about the BLEU calculations, only the paper of S2T-Trans mentioned that they implement SacreBLEU for the BLEU calculation.

As Post(2018), who proposed the SacreBLEU implementation tool, mentioned that, BLEU is in fact a parameterized metric whose values can change dramatically with different parameters and different preprocessing schemes. This information is often not included or hard to find among the research papers, making the direct comparison of BLEU scores between these papers hard. For the parameters, there may be different numbers of references used, computation of the length penalty, maximum n-gram length, and smoothing applied to 0-count n-grams. For pre-processing schemes, different preprocessing schemes such as tokenization, normalization, and compound splitting may be applied to the reference sentences.

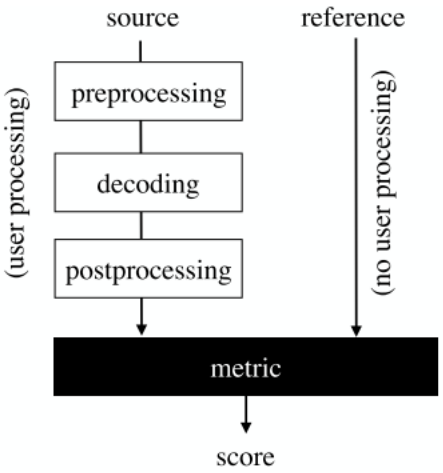

Figure 24 - The proper pipeline for computing reported BLEU scores proposed by Post(2018)

Considering this issue, Post(2018) depicts the ideal process for computing the sharable BLEU scores, shown in Figure 24, and proposed SacreBLEU, which is a Python script that aims to treat BLEU with more reverence by:

- Expecting detokenized outputs as its input due to its own metric-internal preprocessing, producing the same values as WMT
- Automatically downloading and storing WMT and IWSLT 2017 test sets
- Producing a short version string that documents the settings applied

SacreBLEU enables a more shareable, comparable, and reproducible version of BLEU, which has become a standard good practice(Núñez-Marcos et al., 2023).

The implementation details of BLEU scores calculation are unknown for papers of S2(G+T)-Trans, S2T-LCU-Trans, and S2T-GA-Trans, while only that of the S2T-Trans uses the SacreBLEU. Therefore, it is possible that these papers do not use the same BLEU calculation parameters, introducing unfairness to the comparisons and evaluations between the training approaches these papers proposed. For instance, it is unknown how much these BLEU calculation differences may have possibly contributed to that the model trained using S2T-Trans with a deeper network performs better than the others.

8.5.2 Other Evaluation Metrics

The SLT models trained in the experiments of this paper are evaluated and compared only according to the BLEU, especially BLEU-4, scores. Although BLEU has significant advantages and is the most used metric in the literature for MT tasks including the SLT model, it may not be the ultimate measure for improved machine translation quality(Callison-Burch et al., 2006). BLEU has some drawbacks for tasks that rely heavily on contextual understanding and reasoning(Celikyilmaz et al., 2020) since it considers neither semantic meaning nor sentence structure. Especially, Tatman(2019) also mentioned that it does not handle morphologically rich languages well. When there are outliers in machine translation tasks, BLEU leads to high correlations and facilitates false conclusions(Mathur et al., 2020).

Considering the mentioned drawbacks and that sign languages are generally considered morphologically rich languages, it is suggested that BLEU should not be the only evaluation metric for the SLT model. According to Graham(2015), while the previous research papers concluded that BLEU achieves a strong correlation with human assessment, it does not significantly outperform some of the other evaluation metrics in all circumstances.

For instance, ROUGE (Lin, 2004) is another popular evaluation metric besides BLEU. Compared to BLEU, ROUGE focuses more on recall rather than precision, and thus it is more interpretable than BLEU(Callison-Burch et al., 2006). While BLEU uses a simple n-gram match count to measure the precision of the model-generated text, ROUGE uses different types of n-grams, such as skip-bigrams and occurrence statistics(Celikyilmaz et al., 2020). Besides, other evaluation metrics that may also be suitable for SLT tasks such as CHRF, METEOR, and NIST have not been explored in this paper.

Besides, in the paper on S2T-Trans(Tarrés et al., 2023), they proposed an evaluation metric based on BLEU, which is reducedBLEU (rBLEU). Inspired by (Dey et al., 2022), Tarrés et al.(2023) proposed rBLEU as a validation metric for the SLT model trained on the dataset they created. The metric rBLEU consists of removing certain words from the prediction and

the reference before computing the BLEU scores. They created a blacklist of words that frequently appeared in the training dataset but do not contribute much to the meaning of the sentences, such as pronouns, prepositions, and articles. As a result, rBLEU is claimed to be a more accurate evaluation than BLEU. Especially, rBLEU prevents score inflation when low-resource datasets with frequent repetitive patterns are used for training, which is a common case for SLT model training. rBLEU was not included as an evaluation metric in this dissertation.

9 Conclusion

In this dissertation, through an attempt to train a BSL translation model using various Transformer model training approaches and datasets, a brief overview of the current situation of the development of neural machine translation for sign language was conducted. This section does a brief conclusion on findings according to the objectives mentioned in Section 1.2.

Objective 1: Training BSL translation models

The Transformer training approaches proposed by four research papers, including S2(G+T)-Trans, S2T-LCU-Trans, S2T-GA-Trans, and S2T-Trans, are replicated on SL datasets BOBSL and BSLCP. One major reason for selecting these papers is the publicity of their code which makes them replicable. Besides, S2(G+T)-Trans was the first Transformer and state-of-the-art approach for the SLT task while using gloss. Considering training the SLT model without the requirement of gloss is one of the main goals in SLT research, models of the other three papers are proposed to be gloss-free approaches using a conventional Transformer or modification version of the S2(G+T)-Trans approach.

Generally, the inference performance of the trained SLT models is bad. Without any data filtering, the accuracies of SLT models are generally less than BLEU-4 0.8, where models trained on BOBSL achieve performance worse than the models trained on BSLCP. Among the training approaches, the S2T-Trans approach with conventional Transformer training without glosses achieves the best result, while S2T-LCU-Trans is the second, followed by the gloss-supervised training with S2(G+T)-Trans, and the S2T-GA-Trans training without gloss is the worst.

In terms of the training experiments on BOBSL with data filtering applied, it is observed that filtering, based on the degree of synchronization between sign instances auto-spotted and the sentence text, increases the performance of the SLT model. With the filters becoming stricter,

the number of sentences decreases, which in turn makes it harder for the model to learn the meaning and reduces the performance of the model.

Objective 2a: Evaluation of the datasets

First, during the comparison between BSL datasets and the benchmarking dataset PHOENIX-2014t, the importance of large domains of discourse is observed regarding the practicality and robustness of the SLT model. Although linguistic corpus like BSLCP offers SL data with decent quality sign-to-text annotations, they are hard to scale enough for the data-consuming SLT model training. Meanwhile, constructing scalable large-scale interpretation-based datasets like BOBSL is a potential solution to data scarcity in SL research. However, the training results indicated there is still much work required on improving and utilizing that kind of dataset, such as larger amounts of data, denser auto-annotations, more accurate subtitles auto-alignment, and better data filtering methods. Also, sign interpretation is heavily reliance on context to convey meaning from spoken languages, which makes most of the current LST model training approaches that utilize sentence-level data pair as data entry less optimal.

Objective 2b: Evaluation of data pre-processing methods

Besides, the significant effect of the data pre-processing method is observed, for both the sign language video input and the spoken language annotation text. For the visual representations of the SL video, the training results show the importance of selecting and pre-training the features extraction model, especially the I3D feature extraction CNN pre-trained in an isolated sign language recognition setup helps the model to lean a better inference accuracy. The better performance of the SLT model trained with the S2T-Trans paper also indicates that tokenizing the annotations with a compatible spoken language model helps the training.

Objective 2c: Evaluations of the experimented training approaches

Without enough gloss annotations, the SLT model trained with the gloss-supervised approach generally cannot outperform the gloss-free training approach. For the gloss-free approach, it is observed that the SLT model trained using the gloss-free approach with a deeper conventional

Transformer network with regularization performs better, while the other experimented approaches with methods and modifications on the conventional Transformer proposed to help the model learn better without gloss supervision, including stochastic weight, transfer knowledge from spoken language model and gloss attention, did not achieve good results in this dissertation. Despite that, we think that related research is on the track to further enhancing the gloss-free SLT accuracy.

Major limitations of the dissertation

The major limitations of this dissertation are related to the data availability and training approaches. The data scarcity is one of the main stumbling blocks of SL research. This dissertation even manages to use only a partial amount of the BSL data for training. For the attempted training approaches, detailed experiments like ablation tests to evaluate the methods used by each training approach proposed were not conducted, due to the limitations of time and personal capability for significant code modification. The attempted Transformer approaches are limited to those with code publicly available, leaving some of the recently published multi-encoder Transformer approaches not being reproduced, such as an approach that better handles the context for the scalable interpretation-based dataset and the limitation of sentence-pair data.

10 Future Work

This section discusses future work according to the limitations of this paper, other unexplored techniques, and practical application of the SLT technology in the sign language user society.

10.1 Transformer Architecture

In the future, we anticipate the replication and exploration of the multi-encoder or multi-model Transformer architecture, with a particular focus on two proposed multi-model approaches: one for enhancing context awareness(Sincan et al., 2023) and one for multi-model for incorporating visual cues(Shi et al., 2022). Besides, some much explorations could be done on finding better pre-trained decoder for spoken language, like the BER-based model(De Coster & Dambre, 2022).

10.2 Real-Life Application

In addition to experiment approach for training SLT models with better accuracy and robustness, much work is required to bring the related technologies to real-life application. Although the inference accuracy is far from practical for real-time translation in general, it is proposed to build an SLT application for specific domains, such as using the PHOENIX-2014T dataset to train the SLT model for weather domain real-time translation. Also, SL educational platform with SL data collection, with permission in advance, is also a viable future work direction.

11 Learning Reflection

Throughout the course of my dissertation, I obtained substantial personal growth and learned a great deal about the process of finding usable resources and conducting research.

11.1 Personal Growth

In terms of the growth in technical skills, this dissertation allows me to experience the research process of training and modifying AI models for practical usage. My previous projects were mostly related to software engineering, sometimes just applying AI models to achieve some functionalities. During the dissertation, it was my first time training AI models, with data pre-processed and rearranged by myself, attempting to achieve the best performance of the model. It was also my first time conducting a thorough investigation, reading a considerable amount of paper, on the latest development of a specific technology, namely the SLT.

For the knowledge growth, I got familiar with the nature of the sign languages and the challenges their users are facing. I also became acquainted with various structures of the neural network and their practical applications through their application on SL-related tasks. For instance, using CNN as a visual feature extractor, and multi-encoder Transformer for capturing multiple visual cues or video context.

Besides, I found some of my personal strengths and weaknesses. For strengths, I found myself with perseverance in pursuing certain goals, such as reproducing others' works, extracting the code section for visual embedding, and experimenting with various data pre-processing methods and data filters for the improvement of the model's performance. For instance, I devised the video features extraction to be performed in a multi-processing, which makes multiple attempts of training with different data available before deadline. For weaknesses, I have a huge improvement space in English writing skills, especially in expressing the findings in an explicit way and better organizing the sections. I am also far from familiar with the details of the neural network architecture and training implementation in Python, limiting myself from experimenting with different methods applied in training.

11.2 Usability of Resources

In terms of the availability of resources in this dissertation, it was mostly related to the availability of the trained model weights, training parameters, and the code for reproducing the different experiments of the papers with SLT models training approaches proposed. For instance, the pre-trained weights of the 2D CNN for visual feature extraction in the paper on S2(G+T)-Trans(Camgöz et al., 2020) were not provided. Thus, I looked through other papers later released which include other methods for feature extraction. One of the methods I later found and experimented with is a fine-tuned I3D CNN approach. It constituted a state-of-the-art performance for SL visual extraction which is better than the original 2D CNN approach. Through this experience, I learned the advantages of always searching for alternative approaches to the one mentioned in a paper. As a result, I could either substitute the approach with unavailable resources with alternative methods or facilitate myself to grapes primary ideas for comparison of the approaches.

For the authority of sources, I have learned to be aware of being dependent on solely one paper. Instead, I should always search for other papers that cited the paper, or other papers doing a similar task. The paper on the S2T-LCU-Trans(Voskou et al., 2021) training approach claimed that it is a method without the requirement of gloss annotation information. However, later papers comment on this paper, mentioning that it is not actually gloss-free as the gloss information is implicit in the training process. In short, this experience emphasizes the need to seek out multiple sources to ensure accuracy.

11.3 Research Focus Switch

The original plan for the evaluation or discussion of this dissertation focused on the training approaches, especially the Transformer architecture, proposed by the papers. However, due to limitations of time and my capability, I could not make a lot of modifications to the network architecture and training parameters. Thus, in-depth analyses of the training

approaches could not be performed through experiments such as ablation tests and fair comparisons of different techniques applied.

Instead of in-depth analyses of the training approach, I focused more on analyzing and discussing the better utilization of the automatic annotated interpretation-based SL dataset, namely the BOBSL. The dataset is more scalable than the conventional SL datasets, thus better utilization of it is considered a possible solution to the data scarcity in the SLT research development. These findings were not expected at first since I was just aiming to increase the accuracy of the SLT model by applying various combinations of filter settings to the dataset.

Therefore, some research findings may not be expected at the beginning, but during the experiment. A similar situation was observed in a paper by Varol et al.(2021), their original aim was to devise an SLT model. Although the SLT model yields low accuracy inferencing performance, they found the model is useful in automatically temporal localizing the sign instances in SL video, which then significantly increases the annotations of the large-scale dataset BOBSL. In conclusion, this experience highlights the importance of being open-minded and adaptable in the research process and being willing to explore new avenues and possibilities that may arise during the project.

12 References

Albanie, S., Varol, G., Momeni, L., Bull, H., Afouras, T., Chowdhury, H., Fox, N., Woll, B., Cooper, R., McParland, A., & Zisserman, A. (2021). *BBC-Oxford British Sign Language Dataset*. http://arxiv.org/abs/2111.03635

Bahdanau, D., Cho, K., & Bengio, Y. (2014). *Neural Machine Translation by Jointly Learning to Align and Translate* [Article]. https://doi.org/10.48550/arxiv.1409.0473

Bellugi, U., & Fischer, S. (1972). A comparison of sign language and spoken language [Article]. *Cognition*, *1*(2), 173–200. https://doi.org/10.1016/0010-0277(72)90018-2

Bragg, D., Koller, O., Bellard, M., Berke, L., Boudrealt, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., Vogler, C., & Morris, M. R. (2019). *Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective* [Article]. https://doi.org/10.48550/arxiv.1908.08597

Bull, H., Afouras, T., Varol, G., Albanie, S., Momeni, L., & Zisserman, A. (2021). *Aligning Subtitles in Sign Language Videos*. https://www.robots.ox.ac.uk/

Callison-Burch, C., Osborne, M., & Koehn, P. (2006). *Re-evaluating the Role of BLEU in Machine Translation Research*.

Camgoz, N. C., Saunders, B., Rochette, G., Giovanelli, M., Inches, G., Nachtrab-Ribback, R., & Bowden, R. (2021a). *Content4All Open Research Sign Language Translation Datasets* [Article]. https://doi.org/10.48550/arxiv.2105.02351

Camgoz, N. C., Saunders, B., Rochette, G., Giovanelli, M., Inches, G., Nachtrab-Ribback, R., & Bowden, R. (2021b). Content4All Open Research Sign Language Translation Datasets. *Proceedings - 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021*. https://doi.org/10.1109/FG52635.2021.9667087

Cao, Z., Hidalgo, G., Simon, T., Wei, S. E., & Sheikh, Y. (2021). OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(1), 172–186. https://doi.org/10.1109/TPAMI.2019.2929257

Cao, Z., Simon, T., Wei, S.-E., & Sheikh, Y. (2016). *Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields* [Article]. https://doi.org/10.48550/arxiv.1611.08050

Carmen Cabeza, & Jośe M. Garć ıa-Miguel. (2018). *iSignos: Interfaz de datos de Lengua de SignosEspa˜nola (versíon 1.0)*.

Carreira, J., & Zisserman, A. (2017). *Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset*. http://arxiv.org/abs/1705.07750

Celikyilmaz, A., Clark, E., & Gao, J. (2020). *Evaluation of Text Generation: A Survey*. http://arxiv.org/abs/2006.14799

Chen, Y., Wei, F., Sun, X., Wu, Z., & Lin, S. (2022). *A Simple Multi-Modality Transfer Learning Baseline for Sign Language Translation*.

Cihan Camgoz, N., Hadfield, S., Koller, O., Ney, H., & Bowden, R. (2018). *Neural Sign Language Translation*. https://www-i6.informatik.rwth-aachen.de/

Cihan Camgöz, N., Koller, O., Hadfield, S., & Bowden, R. (2020). *Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation*.

CORMIER, K. (2006). Wendy Sandler & Diane Lillo-Martin, Sign language and linguistic universals. Cambridge: Cambridge University Press, 2006. Pp. xxi+547 [Article]. *Journal of Linguistics*, *42*(3), 738–742. https://doi.org/10.1017/S0022226706314387

De Coster, M., & Dambre, J. (2022). Leveraging Frozen Pretrained Written Language Models for Neural Sign Language Translation. *Information (Switzerland)*, *13*(5). https://doi.org/10.3390/info13050220

De Coster, M., Shterionov, D., Van Herreweghe, M., & Dambre, J. (2023). Machine translation from signed to spoken languages: state of the art and challenges. In *Universal Access in the Information Society*. Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/s10209-023-00992-1

De Sisto, M., Vandeghinste, V., Egea Gómez, S., De Coster, M., Shterionov, D., & Saggion, H. (2022). *Challenges with Sign Language Datasets for Sign Language Recognition and Translation*. https://www.corpusvgt.be/

Dey, S., Pal, A., Chaabani, C., & Koller, O. (2022). *Clean Text and Full-Body Transformer: Microsoft's Submission to the WMT22 Shared Task on Sign Language Translation* [Article]. https://doi.org/10.48550/arxiv.2210.13326

Duarte, A., Albanie, S., Giró-I-Nieto, X., & Varol, G. (2021). *Sign Language Video Retrieval with Free-Form Textual Queries*. https://imatge-upc.github.io/sl_retrieval/

Graham, Y. (2015). *Re-evaluating Automatic Summarization with BLEU and 192 Shades of ROUGE*. Association for Computational Linguistics.

Hall, W. C., Levin, L. L., & Anderson, M. L. (2017). Language deprivation syndrome: a possible neurodevelopmental disorder with sociocultural origins [Article]. *Social Psychiatry and Psychiatric Epidemiology*, *52*(6), 761–776. https://doi.org/10.1007/s00127-017-1351-7

Heinzerling, B., & Strube, M. (2017). *BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages* [Article]. https://doi.org/10.48550/arxiv.1710.02187

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, & Li Fei-Fei. (2009). *ImageNet: A Large-Scale Hierarchical Image Database*. IEEE.

Ko, S. K., Kim, C. J., Jung, H., & Cho, C. (2019). Neural sign language translation based on human keypoint estimation. *Applied Sciences (Switzerland)*, *9*(13). https://doi.org/10.3390/app9132683

Koller, O., Camgoz, N. C., Ney, H., & Bowden, R. (2020). Weakly Supervised Learning with Multi-Stream CNN-LSTM-HMMs to Discover Sequential Parallelism in Sign Language Videos [Article]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(9), 2306–2320. https://doi.org/10.1109/TPAMI.2019.2911077

Kudo, T., & Richardson, J. (2018). *SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing* [Article]. https://doi.org/10.48550/arxiv.1808.06226

Landar, H. (1961). Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf [Article]. *Language*, *37*(2), 269–271. https://doi.org/10.2307/410856

Li, D., Rodriguez Opazo, C., Yu, X., & Li, H. (2020). *Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison*. https://dxli94.github.io/

Lin, C.-Y. (2004). *ROUGE: A Package for Automatic Evaluation of Summaries*.

Lin, K., Wang, X., Zhu, L., Sun, K., Zhang, B., & Yang, Y. (2023). *Gloss-Free End-to-End Sign Language Translation*. http://arxiv.org/abs/2305.12876

Loshchilov, I., & Hutter, F. (2016). *SGDR: Stochastic Gradient Descent with Warm Restarts* [Article]. https://doi.org/10.48550/arxiv.1608.03983

Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.-L., Yong, M. G., Lee, J., Chang, W.-T., Hua, W., Georg, M., & Grundmann, M. (2019). *MediaPipe: A Framework for Building Perception Pipelines*. http://arxiv.org/abs/1906.08172

Mathur, N., Baldwin, T., & Cohn, T. (2020). *Tangled up in BLEU: Reevaluating the Evaluation of Automatic Machine Translation Evaluation Metrics*. http://arxiv.org/abs/2006.06264

Momeni, L., Bull, H., Prajwal, K. R., Albanie, S., Varol, G., & Zisserman, A. (2022). *Automatic dense annotation of large-vocabulary sign language videos*. http://arxiv.org/abs/2208.02802

Moryossef, A., & Goldberg, Y. (2021). *Sign Language Processing*.

Müller, M., Ebling, S., Avramidis DFKI Berlin Alessia Battisti, E., Berger HfH Zurich Richard Bowden, M., España-Bonet DFKI Saarbrücken Roman Grundkiewicz Microsoft Zifan Jiang, C., Koller, O., Moryossef, A., Perrollaz HfH Zurich Sabine Reinhard HfH Zurich Annette Rios, R., Shterionov, D., & Sidler-Miserez HfH Zurich Katja Tissi HfH Zurich Davy Van Landuyt, S. (n.d.). *Findings of the First WMT Shared Task on Sign Language Translation (WMT-SLT22)*. https://www.2022.aclweb.org/

Murray, J. J., Hall, W. C., & Snoddon, K. (2020). The Importance of Signed Languages for Deaf Children and Their Families [Article]. *The Hearing Journal*, *73*(3), 30–32. https://doi.org/10.1097/01.HJ.0000657988.24659.f3

Nonhebel, A., Crasborn, O., & Van Der Kooij, E. (2004). *Sign language transcription conventions for the ECHO Project*. http://www.let.kun.nl/sign-lang/echo/docs/transcr_conv.pdf

Núñez-Marcos, A., Perez-de-Viñaspre, O., & Labaka, G. (2023). A survey on Sign Language machine translation. In *Expert Systems with Applications* (Vol. 213). Elsevier Ltd. https://doi.org/10.1016/j.eswa.2022.118993

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). *BLEU: a Method for Automatic Evaluation of Machine Translation*.

Parton, & Becky Sue. (2006). Sign Language Recognition and Translation: A Multidisciplined Approach From the Field of Artificial Intelligence [Article]. *Journal of Deaf Studies and Deaf Education*, *11*(1), 94–101. https://doi.org/10.1093/deafed/enj003

Pfau, Roland., Steinbach, Markus., & Woll, B. (2012). *Sign language : an international handbook* (Roland. Pfau, Markus. Steinbach, & B. (Bencie) Woll, Eds.) [Book]. De Gruyter Mouton. https://doi.org/10.1515/9783110261325

Post, M. (2018). *A Call for Clarity in Reporting BLEU Scores*. http://arxiv.org/abs/1804.08771

Rachel Tatman. (2019). *Evaluating text output in nlp: Bleu at your own risk*. Ttps://Towardsdatascience.Com/Evaluating-Text-Output-in-Nlp-Bleu-at-Your-Own-Risk-E8609665a213.

Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. http://arxiv.org/abs/1908.10084

Samuel Albanie, G¨ul Varol1, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, & Andrew Zisserman. (2020). *BSL-1K - Scaling Up Co-articulated Sign Language*

Recognition Using Mouthing Cues (A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm, Eds.; Vol. 12356). Springer International Publishing. https://doi.org/10.1007/978-3-030-58621-8

Schembri, A., Fenlon, J., Rentelis, R., Reynolds, S., & Cormier, K. (2013). *Building the British Sign Language Corpus*. *7*, 136–154. http://nflrc.hawaii.edu/ldchttp://hdl.handle.net/10125/4592

Sennrich, R., Haddow, B., & Birch, A. (2015). *Neural Machine Translation of Rare Words with Subword Units* [Article]. https://doi.org/10.48550/arxiv.1508.07909

Shi, B., Brentari, D., Shakhnarovich, G., & Livescu, K. (2022). *Open-Domain Sign Language Translation Learned from Online Video*. http://arxiv.org/abs/2205.12870

Sincan, O. M., Camgoz, N. C., & Bowden, R. (2023). *Is context all you need? Scaling Neural Sign Language Translation to Large Domains of Discourse*. http://arxiv.org/abs/2308.09622

Sloetjes Han, & Wittenburg Peter. (2008). Annotation by category-ELAN and ISO DCR. In *6th international Conference on Language Resources and Evaluation (LREC 2008)*.

Stein, D., Schmidt, C., & Ney, H. (2012). Analysis, preparation, and optimization of statistical sign language machine translation [Article]. *Machine Translation*, *26*(4), 325–357. https://doi.org/10.1007/s10590-012-9125-1

Stokoe, W. C. (1980). Sign Language Structure [Article]. *Annual Review of Anthropology*, *9*(1), 365–390. https://doi.org/10.1146/annurev.an.09.100180.002053

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). *Sequence to Sequence Learning with Neural Networks* [Article]. https://doi.org/10.48550/arxiv.1409.3215

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017). *Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning*. www.aaai.org

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014). *Going Deeper with Convolutions* [Article]. https://doi.org/10.48550/arxiv.1409.4842

Tarrés, L., Gállego, G. I., Duarte, A., Torres, J., & Giró-i-Nieto, X. (2023). *Sign Language Translation from Instructional Videos*. http://arxiv.org/abs/2304.06371

Tarrés, L., Gàllego, G. I., Giró-i-Nieto, X., & Torres, J. (2022). *Tackling Low-Resourced Sign Language Translation: UPC at WMT-SLT 22*. http://arxiv.org/abs/2212.01140

Van Herreweghe, Mieke and Vermeerbergen, & Myriam and Demey, E. and D. D. H. and N. H. and V. S. (2015). *Het Corpus VGT. Een digitaal open access corpus van videos and annotaties van Vlaamse Gebarentaal, ontwikkeld aan de Universiteit Gent ism KU Leuven*.

Varol, G., Momeni, L., Albanie, S., Afouras, T., & Zisserman, A. (2021). *Read and Attend: Temporal Localisation in Sign Language Videos*. http://arxiv.org/abs/2103.16481

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need*. http://arxiv.org/abs/1706.03762

Voskou, A., Panousis, K. P., Kosmopoulos, D., Metaxas, D. N., & Chatzis, S. (2021). *Stochastic Transformer Networks with Linear Competing Units: Application to end-to-end SL Translation*.

Xie, S., Sun, C., Huang, J., Tu, Z., & Murphy, K. (2017). *Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-offs in Video Classification*. http://arxiv.org/abs/1712.04851

Yin, A., Zhong, T., Tang, L., Jin, W., Jin, T., & Zhao, Z. (2023). *Gloss Attention for Gloss-free Sign Language Translation*. https://github.

Yin, K., Moryossef, A., Hochgesang, J., Goldberg, Y., & Alikhani, M. (2021). *Including Signed Languages in Natural Language Processing*. http://arxiv.org/abs/2105.05222

Yin, K., & Read, J. (2020). *Better Sign Language Translation with STMC-Transformer*. http://arxiv.org/abs/2004.00588

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2017). *Recent Trends in Deep Learning Based Natural Language Processing* [Article]. https://doi.org/10.48550/arxiv.1708.02709

Zhang, B., Müller, M., & Sennrich, R. (2023). *SLTUNET: A SIMPLE UNIFIED MODEL FOR SIGN LANGUAGE TRANSLATION*. https://github.com/bzhangGo/sltunet.

Zhao, L., Kipper, K., Schuler, W., Vogler, C., Badler, N., & Palmer, M. (2000). A Machine Translation System from English to American Sign Language [Article]. *Envisioning Machine Translation in the Information Future*, *1934*, 54–67. https://doi.org/10.1007/3-540-39965-8_6

Zhou, H., Zhou, W., Qi, W., Pu, J., & Li, H. (2021). *Improving Sign Language Translation with Monolingual Data by Sign Back-Translation* [Article]. https://doi.org/10.48550/arxiv.2105.12397

Zhou, H., Zhou, W., Zhou, Y., & Li, H. (2020). *Spatial-Temporal Multi-Cue Network for Continuous Sign Language Recognition* [Article]. https://doi.org/10.48550/arxiv.2002.03187

13 Appendix

13.1 Multi-model Transformer for Context Awareness

Recently, Sincan et al.(2023) pointed out the major drawbacks of the current straightforward way of using sign language phrase-spoken language sentence pairs to train the SLT models. They considered that sign interpreters heavily rely on the context to understand and convey information from spoken language, given that the vocabulary size of sign language is usually significantly smaller than their spoken language equivalent.

Based on this consideration, they proposed a novel multi-model Transformer architecture, which treats the SLT task in a context-aware manner. They make use of the context from previous sequences and confident predictions to disambiguate weaker visual cues. To achieve this, the Transformer architecture consists of three encoders, including:

- a Video Encoder that captures the frame-level video features,
- a Spotting Encoder that models the recognized sign glosses in the video, and
- a Context Encoder that captures the context of the preceding sign sequences.

The information from these three encoders is combined in a final Transformer decoder to generate the translated text in spoken language. Figure 25 shows an overview of the Transformer architecture.

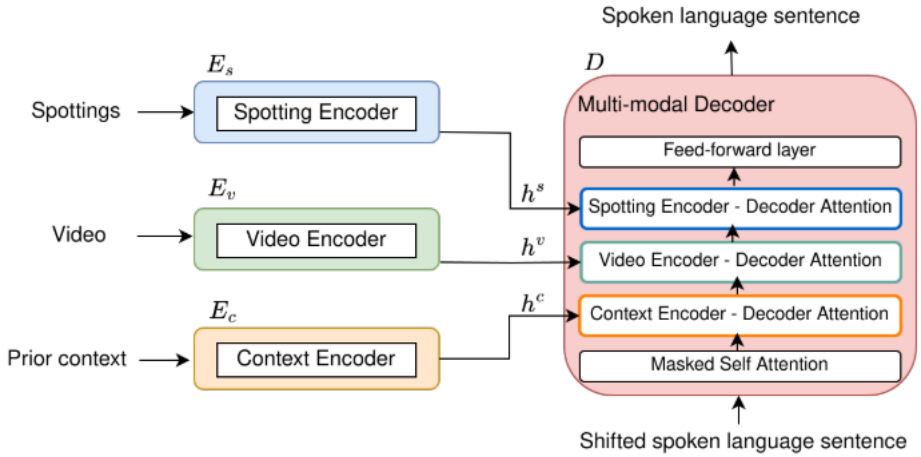

Figure 25 - An overview of the multi-model Transformer architecture(Sincan et al., 2023)

Sincan et al.(2023) conducted a training experiment on the Transformer architecture on the large-scale interpretation-based dataset BOBSL as well, which yielded BLEU-4 scores nearly double the baseline scores. Nonetheless, since they did not publish the code for reproduction, their training was not replicated in this paper due to limited time. The context-aware approach they proposed is capable of better utilization of the signs automatically spotted and the interpretation nature of the subtitles automatically aligned on the large-scale dataset BOBSL, it must be reproduced and further investigated in future work.

13.2 Multi-model Transformer for Visual Cues

In addition to the multi-model Transformer for context-aware translation based on large-scale interpretation-based datasets, mentioned in the previous section, Shi et al.(2022) proposed another multi-model Transformer that guides the SLT model to learn the local discriminative features or visual cues better.

Since sign language usually conveys meaning through a combination of multiple motions, including motions of the mouth, eyebrows, arms, and fingers, the local regions of the frame image play a significant role in distinguishing signs. Although learning the global frame features can include information about the important local cues, achieving this requires a large amount of data(Shi et al., 2022). Considering sign language is a low-resource MT task, Shi et al.(2022) thought it might be helpful to guide the model more explicitly in learning the local discriminative features or visual cues using external tasks.

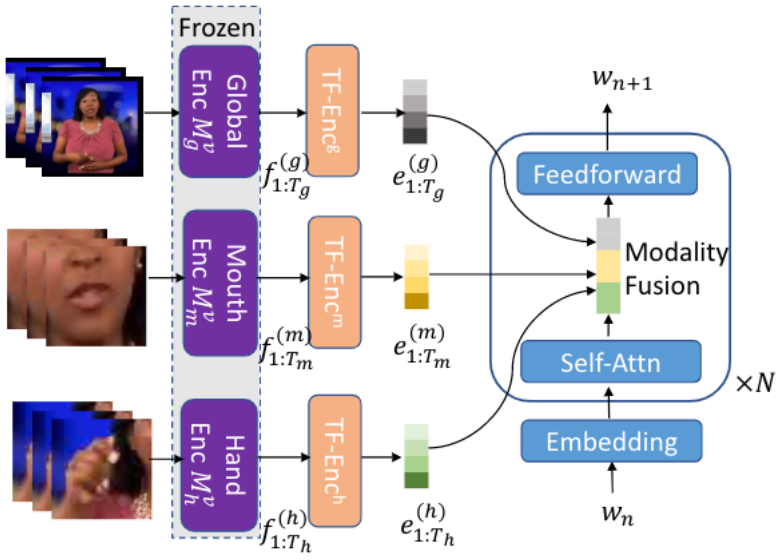

Figure 26 - An overview of the multi-model Transformer architecture(Shi et al., 2022)

They focus on learning features for two local visual modalities, which are handshape and mouthing. They trained a fingerspelling recognizer and implemented an English lip-reading model to extract features from the hand and hand region of interest (ROI), respectively. Similar to the multi-model Transformer for context awareness, their Transformer consists of three encoders, including Global Encoder, Mouth Encoder, and Hand Encoder, and one decoder. An overview of the multi-model Transformer architecture is shown in Figure 26. Hopefully, this Transformer architecture could be reproduced on the interpretation-based dataset BOBSL and further investigated in future works.