UEH University

UEH Institue of Innovation

AI PROJECT FINAL REPORT

Sign Language Translation System Using Machine Learning

Student team:	Team 09
Student name & ID:	Tạ Khánh Hà - 31221023776
	Nguyễn Xuân Hân - 31221023021
	Nguyễn Phương Nghi - 31221020433
	Võ Trần Bảo Trân - 31221024798
	Nguyễn Thị Cẩm Tú - 31221021819
Academic supervisor:	Ph.D Nguyễn Thiên Bảo

Ho Chi Minh City, November 05, 2024

Contributions

No.	Name	Student ID	Percentage of Contribution
1	Tạ Khánh Hà	31221023776	90%
2	Nguyễn Xuân Hân	31221023021	100%
3	Nguyễn Phương Nghi	31221020433	100%
4	Võ Trần Bảo Trân	31221024798	90%
5	Nguyễn Thị Cẩm Tú	31221021819	80%

Contents

Abstract

The main goal of this project is to create a tool that can translate sign language into languages to help people who don't know sign language communicate with the community more easily. Currently the project is concentrating on English, German and French. The website is set up as a user translation platform, like Google Translate that allows for translations in both directions. This project uses machine learning and computer vision technologies to identify and translate sign language gestures into text and vice versa making communication smoother, for everyone involved. We strive not to offer a means, for translating sign language but also to foster inclusivity and equality, in sharing information among all individuals.

1. Introduction

1.1. Project Objectives

The objective of this project is to develop a multilingual sign language translation tool, similar to Google Translate. Its main supported languages will include English, German, and French. Users will be able to input text on which the tool can respond with a video in sign language, hence helping non-signers and the deaf interact better. It will also facilitate translation from sign language to text via video uploads, hence making the platform more accessible for the deaf to present their messages.

1.2. Rationale for Selecting the Topic

Advancements in automated translation technology have made it easier for millions of people to break down language barriers and interact using tools like Google Translate. However, a crucial aspect that hasn't received focus in today translation tools is sign language. This unique form of communication plays a role for the community in engaging with society and sharing their feelings and ideas. The lack of sign language translation tools has posed challenges for individuals with hearing impairments when communicating with those who're not familiar with sign language. Many people who want to learn and comprehend sign language encounter difficulties because of the availability of convenient learning materials. This emphasizes the importance of having a tool that can translate sign language to written language and vice versa.

The project aims to provide numerous benefits and contributions to society; it will serve as a crucial tool for communication between the deaf and the broader community, reducing reliance on interpreters and fostering an inclusive communication environment. Furthermore, the tool will offer significant value for education and research, enabling individuals to easily access and learn sign language. Ultimately, the project will contribute to the establishment of a society free from language barriers, where sign language is widely recognized, thereby promoting linguistic diversity and equality in communication.

2. Literature Review and Related Works

Sign language translation (SLT) facilitates communication between deaf and hearing individuals by converting sign language into spoken or written language. There are two primary approaches in SLT: gloss-based and gloss-free methods. Gloss-based methods use intermediate representations called glosses-textual annotations of sign language movements-while gloss-free methods aim for direct translation from sign language videos to text without intermediate annotations. This review discusses prominent models in both approaches, highlighting their methodologies, strengths, weaknesses, and citing relevant research papers.

2.1. What is Gloss?

Before delving into the two main types of models in sign language translation, we need to learn about the concept of "gloss" in sign language. Glossing is a method used in sign language linguistics and translation to represent the signs in a textual format. Glosses are shorthand approximations that use words from the dominant culture's spoken or written language to translate each sign's meaning; they usually do not express the complete grammar, syntax, or subtleties of the original signed language. Glosses are a useful tool for recording sign languages because they offer a written representation that both hearing people and non-signers may easily understand. However, because glosses frequently fail to capture distinctive structural components like facial expressions, spatial relationships, and classifiers crucial to signed communication, they are unable to accurately describe sign languages [12].

A typical gloss represents each symbol operation with a simplified word or phrase written in capital letters, separated by spaces, and may include symbols or shorthand for non-standard symbols, movements such as moving your eyebrows or tilting your head. For example, the gloss for the phrase “My name is Barbara” might appear as “ME NAME fs-B-A-R-B-A-R-A”.

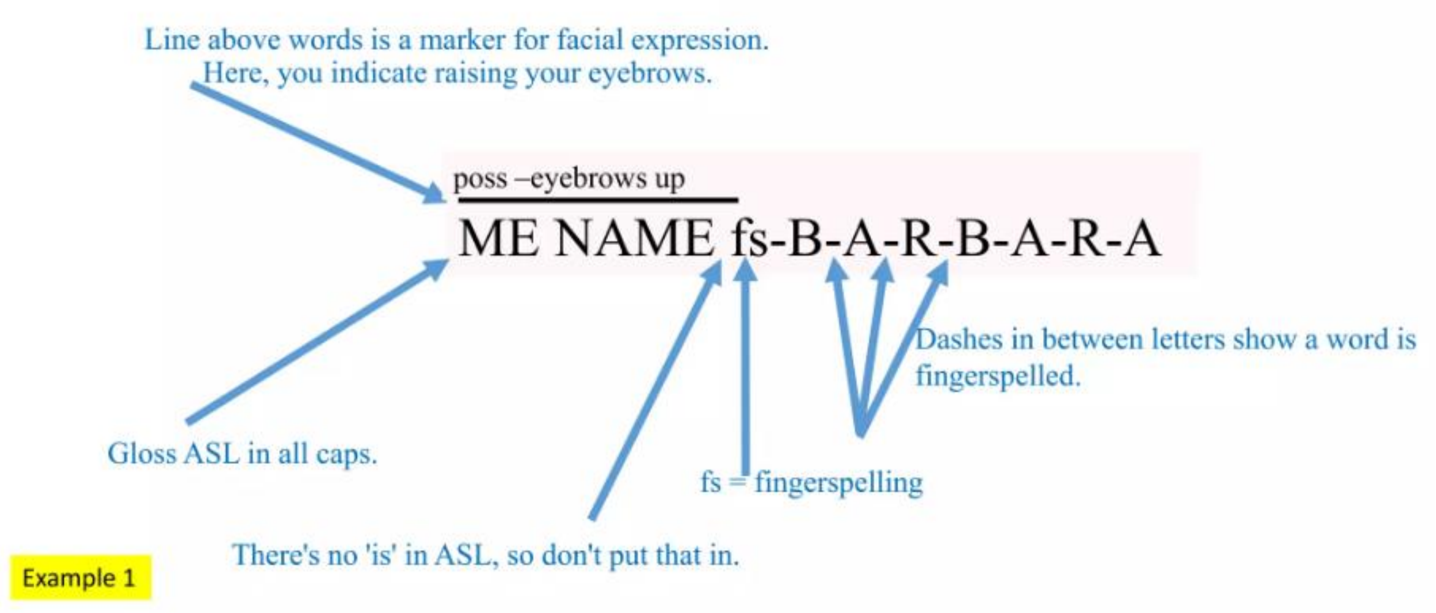

Line above words is a marker for facial expression.
Here, you indicate raising your eyebrows.
poss –eyebrows up
ME NAME fs-B-A-R-B-A-R-A
Dashes in between letters show a word is fingerspelled.
Gloss ASL in all caps.
fs = fingerspelling
There's no 'is' in ASL, so don't put that in.
Example 1

Figure 1. *Example of "glossing" in Sign Language [11]*

Notwithstanding its usefulness, glossing presents difficulties for machine translation of sign languages, especially as glosses are essentially incapable of fully capturing the grammatical structure of signals. Glosses can cause ambiguity and meaning loss in automated translation models because they limit signed sentences to a single lexical item from a spoken language [13]. These drawbacks highlight how crucial it is to investigate alternate representations, like SignWriting or direct feature extraction from video data, as these could provide a more accurate depiction of the expressiveness and grammatical complexity present in sign languages [14].

2.2. Gloss-Based Sign Language Translation Models

Table 1. *Gloss-Based Sign Language Translation Models*

Model name	**Description**	**Reference**
1. Neural Sign Language Translation [4]	Camgoz et al., 2018 introduced a model using CNNs and Bi-LSTM networks for Sign2Gloss and an attention-based Seq2Seq model for Gloss2Text. Strengths include structured learning and improved accuracy, while weaknesses involve annotation dependence and error propagation.	Camgoz, N. C., Koller, O., Hadfield, S., & Bowden, R. (2018). Neural Sign Language Translation. CVPR, 7784-7793.
2. Sign Language Transformer [5]	Yin et al., 2020 introduced a transformer-based model leveraging gloss annotations. It includes Transformer Architecture (self-attention for long-range dependencies) and Multi-Task Learning (joint training for recognition and translation). Strengths are enhanced contextual understanding and joint optimization. Weaknesses include high computational and data demands.	Yin, S., Xia, Z., Chen, X., Zhou, H., & He, S. (2020). Sign Language Translation with Transformer. MM '20, 1778–1786.
3. Neural Sign Language	Orbay & Akarun, 2020 introduced a model that learns sub-word tokenization of glosses, with Tokenization Mechanism	Orbay, E., & Akarun, L. (2020). Neural Sign Language

Translation by Learning Tokenization [6]	and End-to-End Training for efficiency. Strengths are vocabulary efficiency and flexibility. Weaknesses include added complexity and dependence on gloss annotation quality.	Translation by Learning Tokenization. arXiv:2004.03519.

2.3. Gloss-Free Sign Language Translation Models

Table 2. *Gloss-Free Sign Language Translation Models*

Model name	**Description**	**Reference**
1. Sign Language Transformers [7]	Camgoz et al., 2020 developed a gloss-free transformer model with End-to-End Architecture (no gloss representation) and Spatial-Temporal Encoder (CNNs and transformers for video input) plus a Transformer Decoder for target text. Strengths are no gloss annotations needed and nuanced capture of sign input. Weaknesses are data intensiveness and training complexity.	Camgoz, N. C., Hadfield, S., Koller, O., & Bowden, R. (2020). Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation. CVPR, 10023-10033.
2. Progressive Transformers [8]	Saunders et al., 2020 proposed a progressive transformer model with Progressive Learning (intermediate supervision) and Multi-Cue Integration (manual and non-manual cues). Strengths include enhanced feature learning and rich contextual information. Weaknesses are increased model complexity and high resource demand.	Saunders, B., Camgoz, N. C., & Bowden, R. (2020). Progressive Transformers for End-to-End Sign Language Production. ECCV, 687–705.
3. Self-Supervised Learning for SLT [9]	Pu et al., 2019 explored self-supervised learning for gloss-free SLT using Cross-Modal Training (pre-training on related tasks) and Feature Extraction without labeled data. Strengths are data efficiency and improved generalization. Weaknesses include limited improvement without large datasets and additional design complexity.	Pu, J., Zhou, P., Wang, F., & Xu, W. (2019). Boosting Continuous Sign Language Recognition via Cross Modality Augmentation. CVPR, 11509-11518.

About Data Requirements: Gloss-based models can perform well with smaller datasets if gloss annotations are available. Gloss-free models require larger datasets but eliminate the need for costly annotations.

About Model Complexity: Gloss-free models are generally more complex due to their end-to-end nature and need for extensive feature extraction.

About Performance: While gloss-based models currently have an edge in performance due to structured intermediate representations, gloss-free models are rapidly improving with advances in deep learning techniques.

About Flexibility and Scalability: Gloss-free models are more adaptable to different sign languages and can scale better in multilingual contexts.

Both gloss-based and gloss-free models have made significant contributions to the field of sign language translation. Gloss-based models offer high accuracy when gloss annotations are available but are limited by data scarcity and potential loss of nuanced information. Gloss-free models present a scalable alternative that can leverage larger datasets without annotations, though they require more data and computational resources. Future research may focus on hybrid models that combine the strengths of both approaches or explore advanced self-supervised learning techniques to enhance gloss-free models.

3. Methology

The authors believed it was important to conduct a deep study of the architectural system for the sign language translation tool, given that existing models and products had many limitations. The architecture designed is described in detail in Section 3.1, with the objective of optimizing performance toward improving the translation and user experience in raising conditions of communication to a higher level in accord with the needs of both the deaf and those who do not know sign language.

3.1. System Architecture Overview

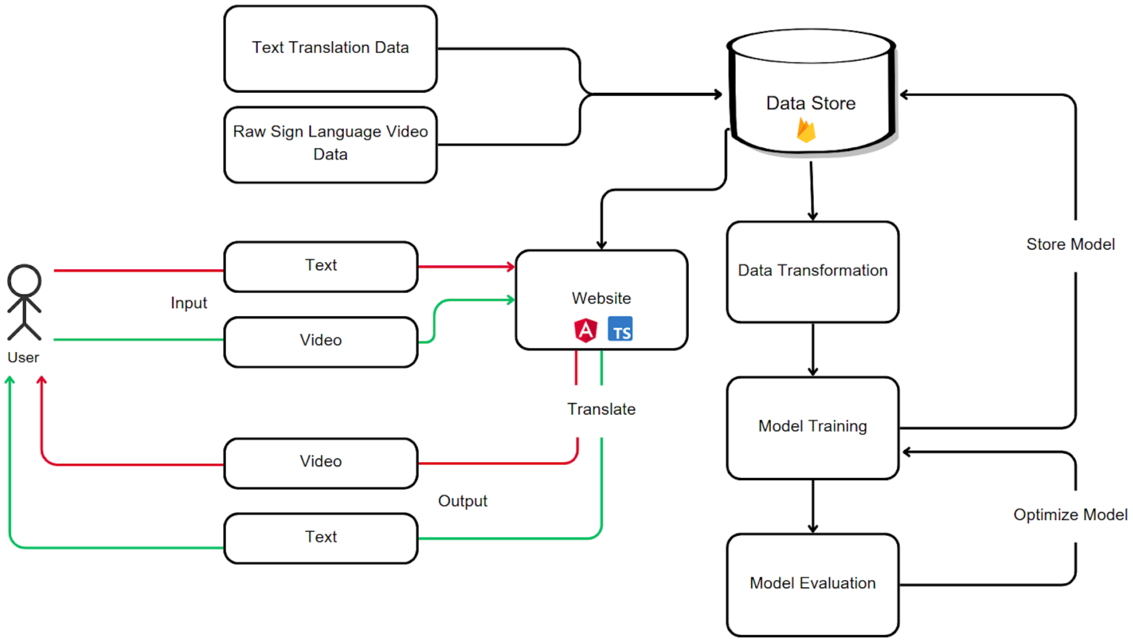


Figure 2. *System Architecture*

A user will input text or video in sign language and after processing the data, this system will return results through the website interface. There are two major sources of data: the text translation data, and the raw data in the form of sign language videos. These are to be stored in a Data Store for maintenance and management of all data, including the versions of the different models trained.

The stage of Data Transformation will convert raw data into a format suitable for training models, and this ensures that translation models receive uniform standardized data. These models will later be trained in the Model Training stage for developing the capability to translate sign languages from texts and videos accurately. Afterwards, the models undergo Model Evaluation & Optimization to perform better with higher accuracy. Optimized models shall be stored at the Data Store, ready to be used in translation processes. Web interface serves as a bridge between users and the system, whereby users input their request and obtain translation results in either text or sign language video format.

3.2. AI model Architecture Overview

- **Approach 1**

Our first approach was to improve on the Stanford students' research [15], which made use of the MS-ASL dataset and the I3D model. Our objective was to apply a Sign2Gloss2Text technique in order to improve this existing model. But while developing, we found 2 problems. First, we came into a few problems when we first downloaded the MS-ASL dataset. The vocabulary data was provided in the JSON files that contained the train, test, and validation sets. With start and end times indicating the section of the video for the intended sign, each word entry matched a link to a YouTube video. However, we had several difficulties when downloading them. Despite the training set JSON including more than 16,000 words, only roughly 1,600 words were downloaded successfully. After examining the dataset, we found that a large number of the YouTube links were either private or no longer accessible. Furthermore, it was often difficult to access parts because the download script pulled the full video rather than just the relevant sections.

Simultaneously addressing the dataset challenges, we commenced the development of the Sign2Gloss2Text model. Although we had no serious problems during the Sign2Gloss phase, we quickly discovered that the Gloss2Text stage was quite difficult. The model needs a dataset with gloss-to-text mappings for this component to function, yet MS-ASL does not offer this dataset. This delays us from completing the Sign2Gloss2Text pipeline as intended.

To solve this problem, we adopted a second approach, using a basic Transformer seq2seq encoder-decoder model with the How2Sign dataset.

- **Approach 2**

In the second method, we translated sign language to spoken language (Sign2Text) directly without the need for a gloss stage using a simple Transformer encoder-decoder paradigm. This model uses a transformer architecture trained on video features retrieved by a pretrained I3D network, as explained in Sign Language Translation from Instructional Videos (CVPR 2023) [16]. In order to produce appropriate English text, the transformer processes the spatial and temporal characteristics that the I3D model extracts from continuous sign language videos.

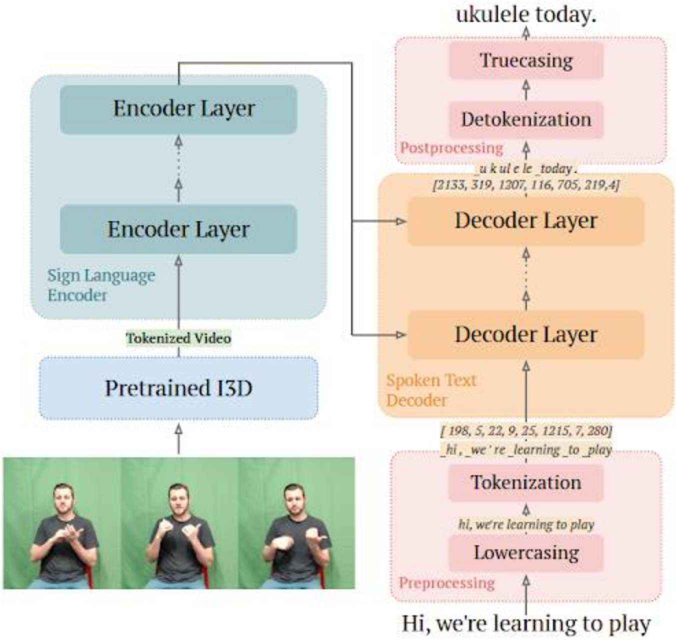


Figure 3. *The input video sequence is fed into a Transformer to generate the output text sequence [16].*

The architecture of this model operates by first passing video data through a pretrained I3D model to extract features, with the output saved as numpy files. These files are then processed by two Transformer layers: an encoder and a decoder, which generate the final translation.

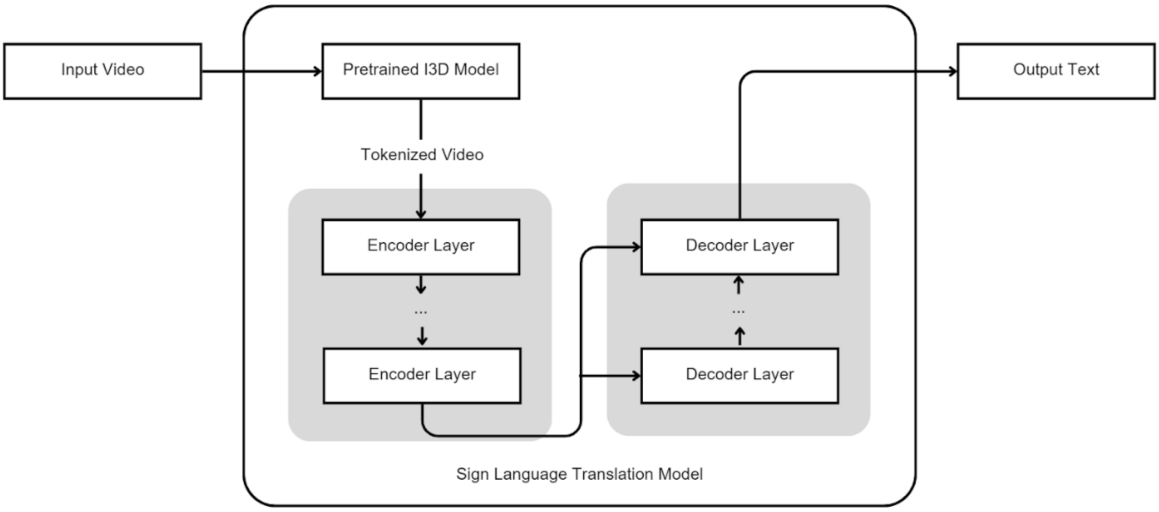


Figure 4. *Our Approach 2 Model based on the Transformer model of [16].*

The advantage of this model design is that it improves contextual awareness of the generated sentences by concentrating on the most important portions of the video sequence during translation. Our findings, however, point to two important drawbacks with this paradigm. First, despite improvements to the model and dataset, translation results were not as encouraging as expected due to low BLEU-4 scores. The BLEU-4 score plateaued at about 8 and the CHRF score can not pass 10 after multiple training epochs, which is insufficient for useful, real-world applications. The difficulty of comprehension of sign language and the model's inability to comprehend lengthy or complicated words account for this. The duration of the training time is the second major barrier. With a single NVIDIA GeForce RTX 3050 Ti GPU, our configuration takes about 18 hours to run 108 epochs, which slows down the iteration process for testing and improving model variations. This restriction makes it more difficult for us to effectively improve the model and test various configurations.

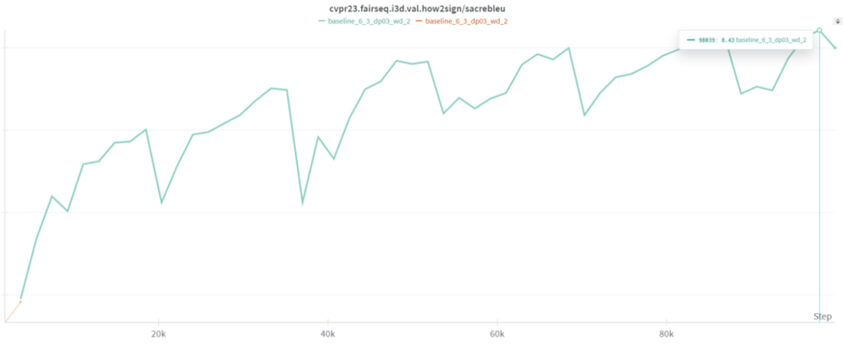


Figure 5. *Approach 2 Model's BLEU-4 Score*

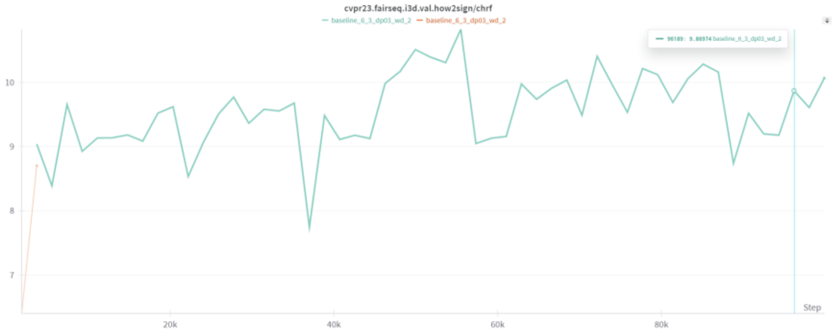


Figure 6. *Approach 2 Model's CHRF Score*

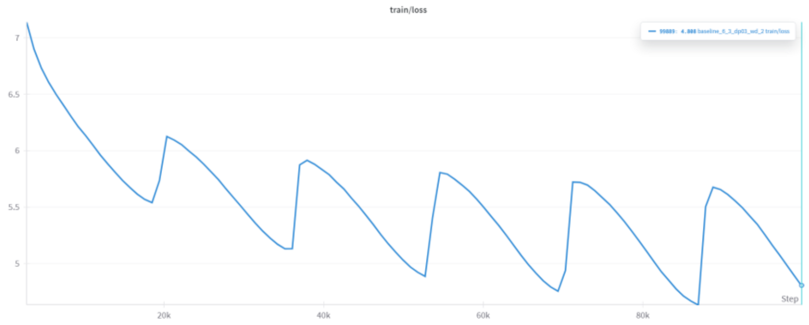


Figure 7. *Approach 2 Model's Loss*

- **Approach 3: Sign2Signwriting2Text and Text2Signwriting2Sign**

To solve the limitations encountered in translating directly without going through gloss as above, we applied a different approach to our project: using SignWriting as an intermediary in the translation process. Unlike gloss, which often simplifies the nuances of sign language, SignWriting is a specialized writing system that represents sign language more precisely. To increase translation accuracy, our final approach - Sign2Signwriting2Text and Text2Signwriting2Sign - use pre-trained models [3].

The system can provide accurate and nuanced translations because of these pre-trained models, which also allow the system to comprehend and express complicated sign language structures and meaning changes. Furthermore, by employing SignWriting, we minimize translation errors and eliminate the ambiguities frequently found in gloss-based systems, enabling a deeper comprehension of the source sign language. Improved performance, particularly higher BLEU-4 scores than earlier models, is one of the key advantages of this technique and shows how well our model maintains translation integrity. In Section 4, we go into great length on the model training procedure, for a thorough examination of this approach and how it affects translation accuracy, and Section 5 on metrics efficiency.

4. Experiment

4.1. System

TypeScript and the Angular framework are used in the frontend interface's construction to offer a user experience that is effective, responsive, and maintainable. When creating apps that are intended to handle real-time input and processing, TypeScript, a superset of JavaScript, helps developers to write highly typed code that is easier to maintain and less likely to make mistakes. Google maintains Angular, a robust front-end framework with a component-based architecture that lets developers grow and modularize apps as needed. We used Angular to create a simple, well-organized user experience that makes navigating easy and allows users to obtain translations with little learning curve.

We included Transloco, an internationalization (i18n) module for Angular that facilitates flexible localization and real-time language translation, to improve accessibility. By smoothly alternating between languages, Transloco's capabilities enable multilingual users and increase the system's usefulness for a wider range of users. This design decision increases the system's worldwide usability by guaranteeing that the interface is inclusive of users who might prefer alternative languages in addition to the Deaf population.

The backend, which is TypeScript-built and Firebase-powered, manages the intricate video processing jobs that make sign language recognition possible. For a system that must process and produce results quickly, Firebase, which is also managed by Google, provides a full range of backend services, such as hosting, real-time data handling, and secure storage. The AI models required for translation are stored in Firebase Storage, allowing the system to quickly and easily retrieve the most recent models.

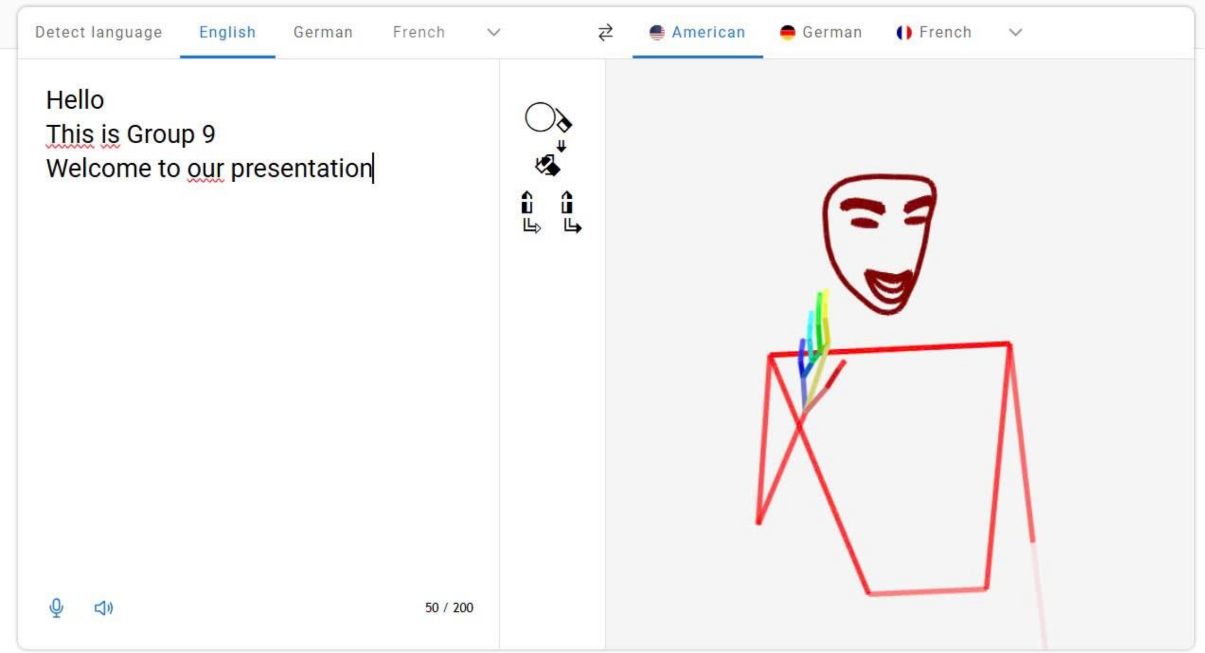
Additionally, we employed Google Research's MediaPipe Holistic machine learning system to extract hand, face, and body landmarks in real time from video input. Because sign languages frequently rely on subtle hand and facial movements to convey context and grammar, MediaPipe Holistic is made to gather extensive gesture and posture data by identifying and mapping key points on the body, face, and hands. These features are crucial for translating sign language. The system can identify movements and give more exact translations by analyzing and translating sign language gestures with high accuracy utilizing MediaPipe Holistic.

The elimination of login and database requirements comes from the goal of maximizing usability and minimizing friction for users. A translation tool needs to be fast, agile, and immediately provide what users need, so the need to create an account to use it can be an obstacle for users. If users want to review or archive the translation, users can directly save their translation by clicking the “Save” button, which will download a zip file containing the .txt file and video file. This local download feature grants users autonomy over their data and simplifies workflow, making it a practical choice for users in a variety of environments, from research personal to real-time conversations.

4.2. AI Architecture

4.2.1. Introduction to Signwriting

As a middle stage, to create the sign language translation task using a sign language writing system (shown in Figure 8): We suggest converting spoken language text into written sign language and then turning this intermediate output into a final video or posture output and in reverse. In this work, we examine the translation between spoken and written signed languages using this multi-step approach of SLT. The intermediate writing system is called SignWriting.



Hello
This is Group 9
Welcome to our presentation

Figure 8. *Demo application based on our models, translating from spoken languages to signed languages represented in SignWriting, then to Skelton poses.*

- **Sign Writing**

SignWriting (Sutton, 1990) is a featural and visually iconic sign language writing system. Previous work explored recognition (Stiehl et al., 2015) and animation (Bouzid and Jemni, 2013) of SignWriting. SignWriting has many advantages, including being computer assisted, universal (multilingual), very simple to learn, and extensively documented. Furthermore, even though it appears pictographic, the writing system is clearly defined. Each sign's location on a two-dimensional plane and a series of symbols (box markers, graphemes, and punctuation marks) can be written down.

SignWriting has two computerized specifications, Formal SignWriting in ASCII (FSW) and SignWriting in Unicode (SWU). SignWriting is two-dimensional, but FSW and SWU are written linearly, similar to spoken languages. Figure 5 gives an example of the relationship between SignWriting, FSW, and SWU. In light of the article we cited, we also concentrate on FSW. [1]

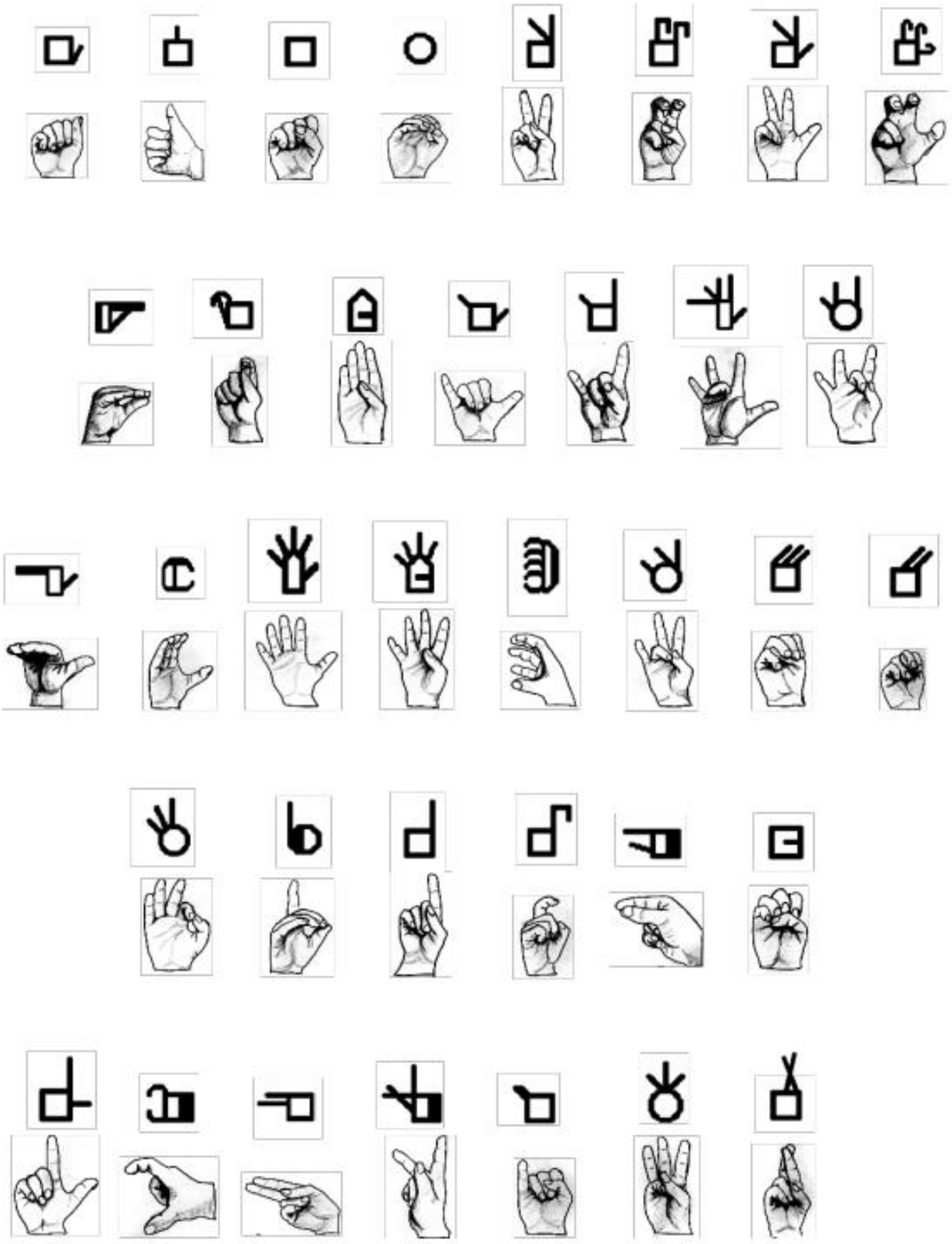


Figure 9. *Hand shapes and their equivalents in SignWriting. [1]*

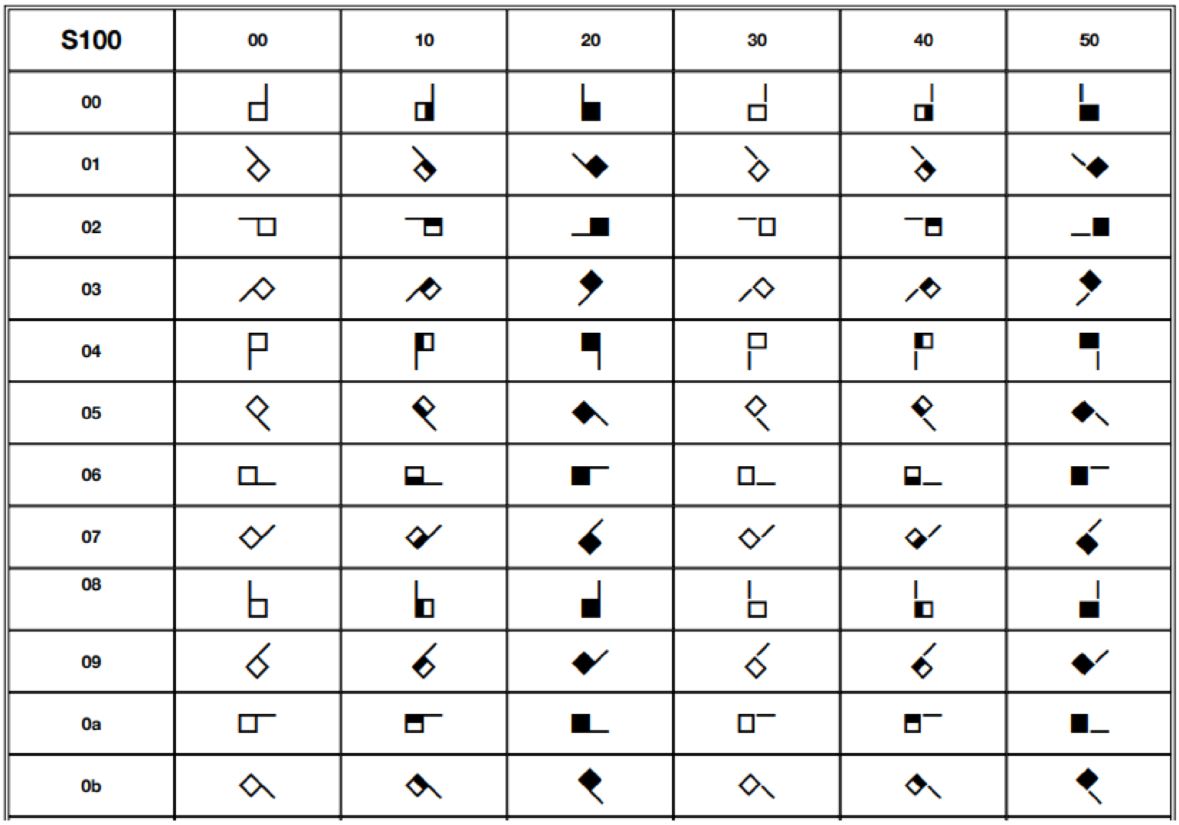


Figure 10. *Orientation of a symbol in SignWriting in 3D space. Each row applies a rotation of the palm in a 2D space vertical to the ground. Each column applies a rotation of the palm in a 2D space parallel to the ground. [1]*

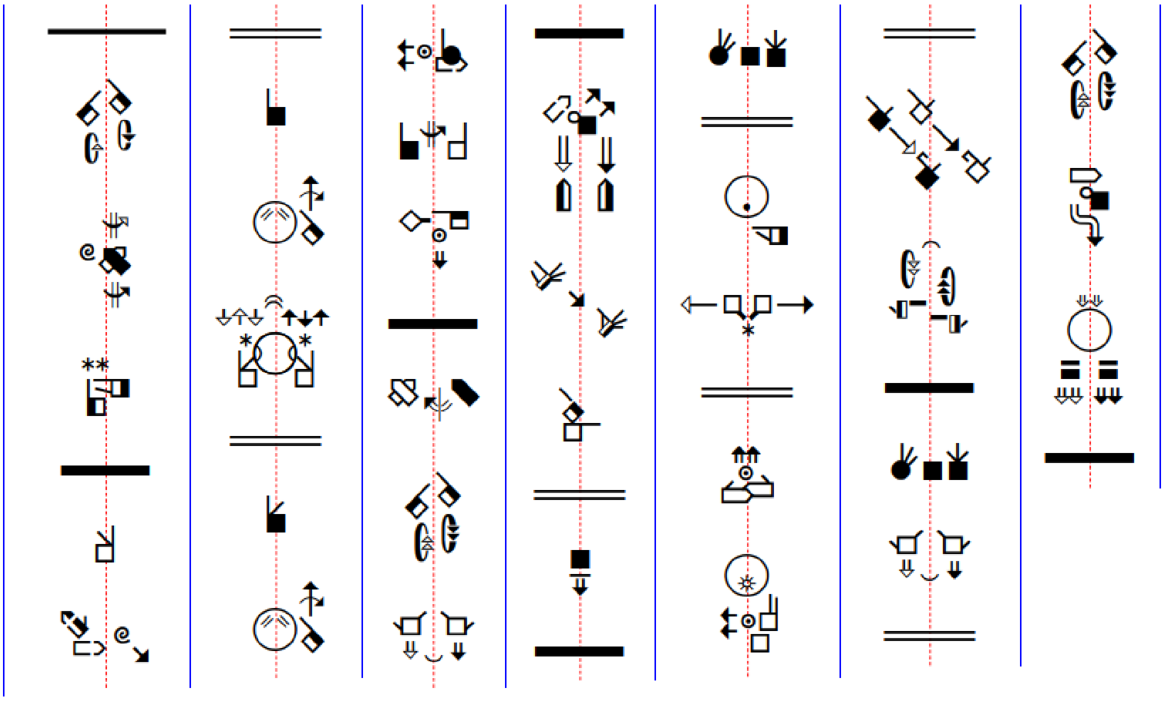
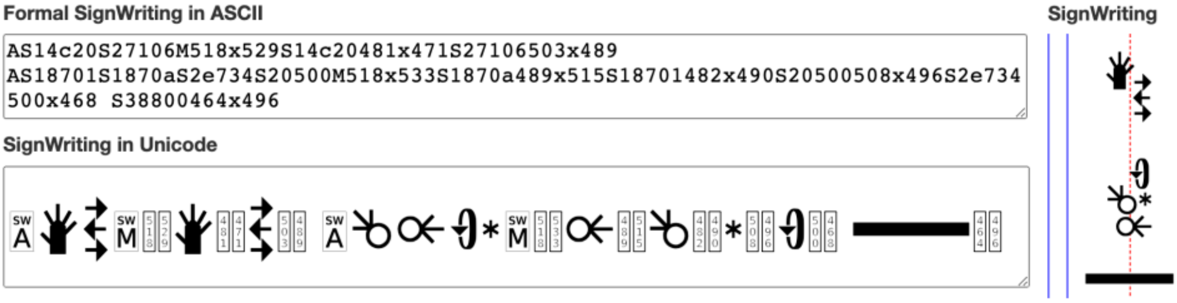


Figure 11. *An example of SignWriting written in columns, ASL translation of an introduction to Formal SignWriting in ASCII. [1]*



Formal SignWriting in ASCII

```
AS14c20S27106M518x529S14c20481x471S27106503x489
AS18701S1870aS2e734S20500M518x533S1870a489x515S18701482x490S20500508x496S2e734
500x468 S38800464x496
```

SignWriting in Unicode

Figure 12. *"Hello world." in FSW, SWU, SignWriting graphics. [1]*

We use the capabilities of Amit Moryossef's pretrained model [2] to produce hand shapes and facial features that enhance the visual components of SignWriting in order to generate SignWriting symbols. By identifying and replicating these particular characteristics, our methodology generates structured outputs in a SignWriting-compatible way.

To identify and encode important hand and facial traits, we would enter video or position data, which the model would then process. It is therefore possible to transfer these encoded symbols into FSW or SWU format SignWriting representations. We can automate a large portion of the symbol creation process by using this model as a foundation, freeing up time to improve the arrangement, placement, and relative orientation of symbols to guarantee a precise and readable SignWriting output.

4.2.2. Details on how to build the model

Dataset: The data source used pre-trained models is SignBank+ [10].

Model:

- **Text2Signwriting2Text:**

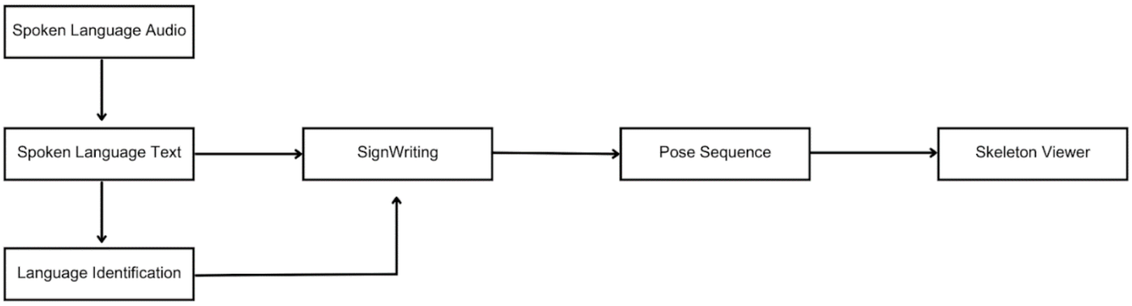
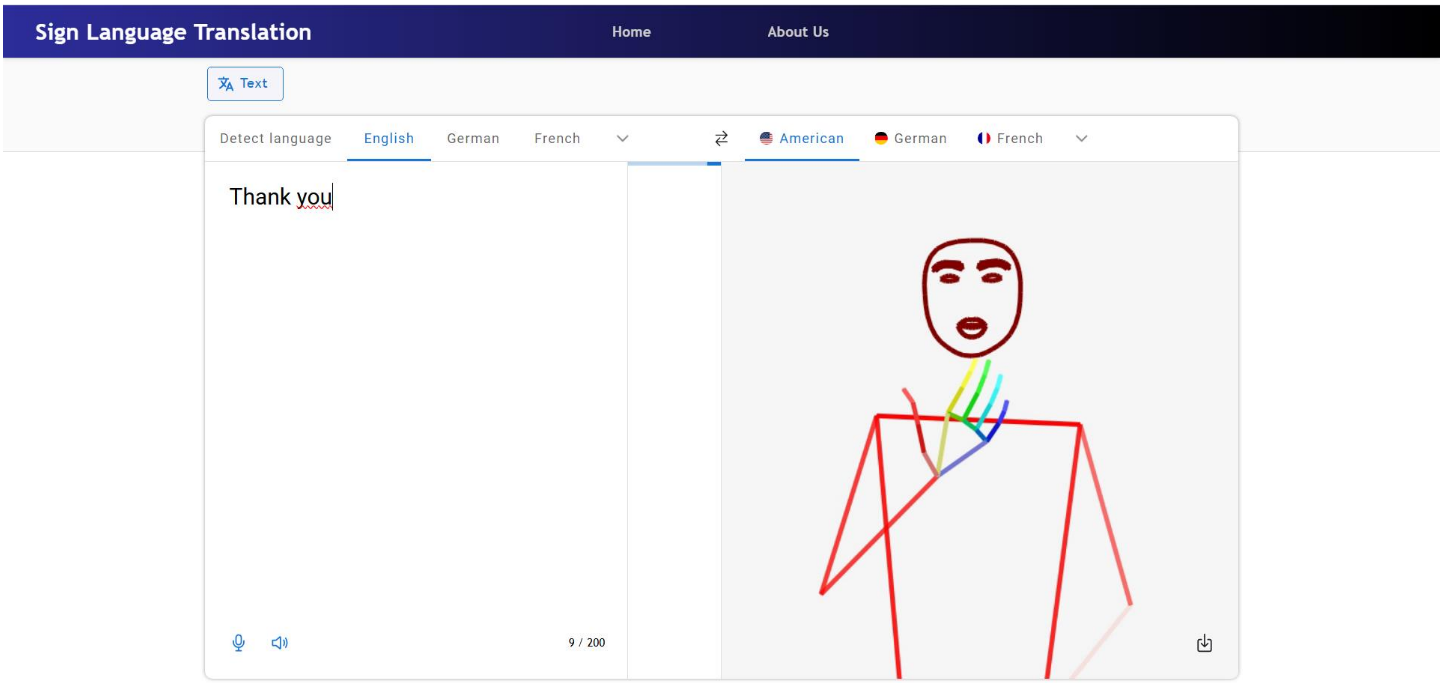


Figure 13. *Spoken Language into Sign Language (The Sign Language Video presents in Skeleton Viewer)*

This model illustrates the way spoken language text and audio can be translated into sign language expression. Text and audio inputs in spoken language are used first. In order to ensure accurate processing, Language Identification assists in identifying the language of the input text or voice. After the spoken language text has been discovered, it is converted into SignWriting, a written method that graphically depicts sign language. A Pose Sequence, which specifies the precise hand and body motions needed to express each sign, is created from this SignWriting output. A Skeleton Viewer, a program that shows a virtual skeleton performing the translated sign language, is then used to visualize these pose sequences.



Thank you

Figure 14. *Example of the input and output of Text2Signwriting2Sign.*

- **Sign2Signwriting2Text**

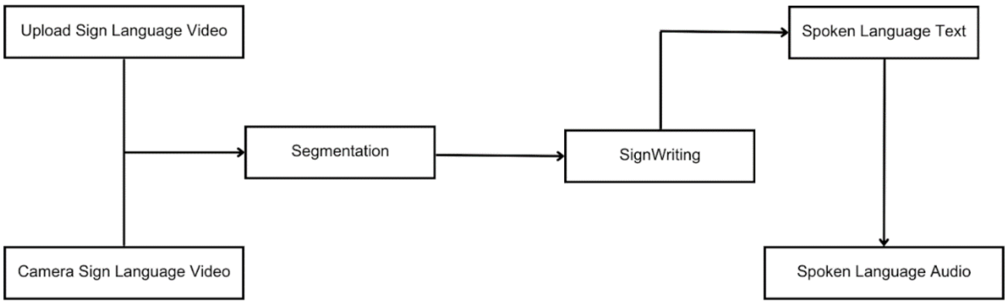


Figure 15. *Sign Language video into Spoken Language.*

By defining the process for converting videos in sign language into spoken language text and audio output, this model allows communication between sign language users and non-signers. There are two choices for entering sign language v ideo data at the start of the procedure. Users have the option to stream live using a webcam that records real-time signing or upload a pre-recorded video in sign language. In the following step, known as segmentation, the model examines the video to separate and recognize distinct signs or sentences. This segmentation stage is essential because it allows the system to separate specific sign language elements from the video frames, readying them for precise translation.

These sign components are converted into SignWriting symbols after they have been segmented. SignWriting is a visual writing system created especially for sign languages that includes facial expressions, hand shapes, and gestures. This model converts visual data into a structured symbolic form by using Amit Moryossef's pretrained model to assist in generating the hand forms and facial features in the SignWriting format. A language model that recognizes and interprets SignWriting as text makes it easier to translate these symbols into spoken language text.

This entire process improves communication accessibility across language modes by ensuring that sign language inputs, whether from live broadcasts or video uploads, are available in spoken language form through both text and audio.

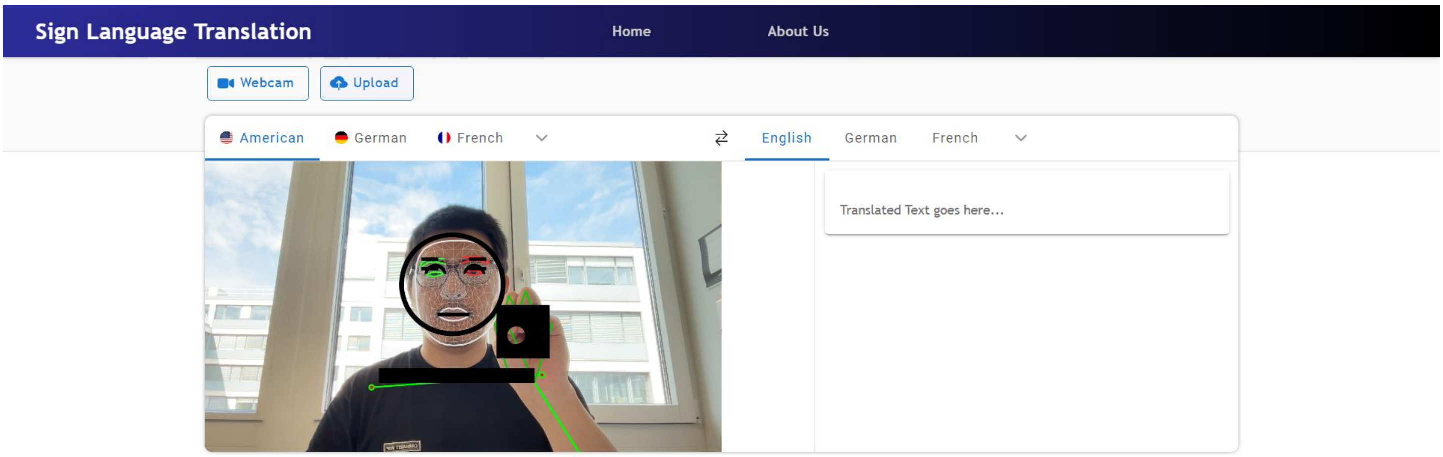


Figure 16. *Example of the input and output of Sign2Signwriting2Text.*

5. Results and Analysis

5.1. Metrics

In the field of machine translation, the BLEU-4 and CHRF measures play an important role in assessing the accuracy of translation models, including the models from text to symbol language (text2sign) and from the symbol language to the text (sign2text).

5.1.1. BLEU-4

BLEU (Bilingual Evaluation Understudy) is one of the most common measures to evaluate machine translation models. The BLEU-4 version, focusing on 4-grams, measuring the accuracy of the phrases consisting of 4 consecutive words between the translation of the model and the reference data. Specifically, BLEU-4 compares the matching of the 4 words phrases, reflecting the accuracy of context and meaning in the translation. BLEU-4 score can be represented from 0 to 1 or from 0 to 100 if calculated by the percentage; High score shows good matching between model and original translation.

In the application of the BLEU-4 for Text2Sign, this measure helps assess the ability of the model to accurately convert important phrases from text to symbols, while maintaining the core meaning. For sign2text, BLEU-4 assesses the accuracy when the symbol language conversion model returns into text, especially in key phrases.

5.1.2. CHRF

CHRF (Character F-Score) is a measure based on characters, developed to meet the requirements of evaluating languages with complex structures or containing shortened elements, such as symbol language. The CHRF is based on the CHRF2 ++, the advanced version of the CHRF, evaluating the accuracy and coverage at the character level, bringing accuracy to the structure of the translation. The CHRF measure is especially useful for sign language because it has the ability to capture small differences in structure, expressed through each specific symbol or sign.

In the application of CHRF for Text2Sign, this measure helps determine the similarity between the output of the model and the original, thereby supporting the accuracy assessment of each specific symbol. For sign2text, the CHRF measures the model's ability to accurately decode each symbol, ensuring that the semantics of the sign language are fully and accurate.

5.2. Results

5.2.1. Text2Signwritng2Sign

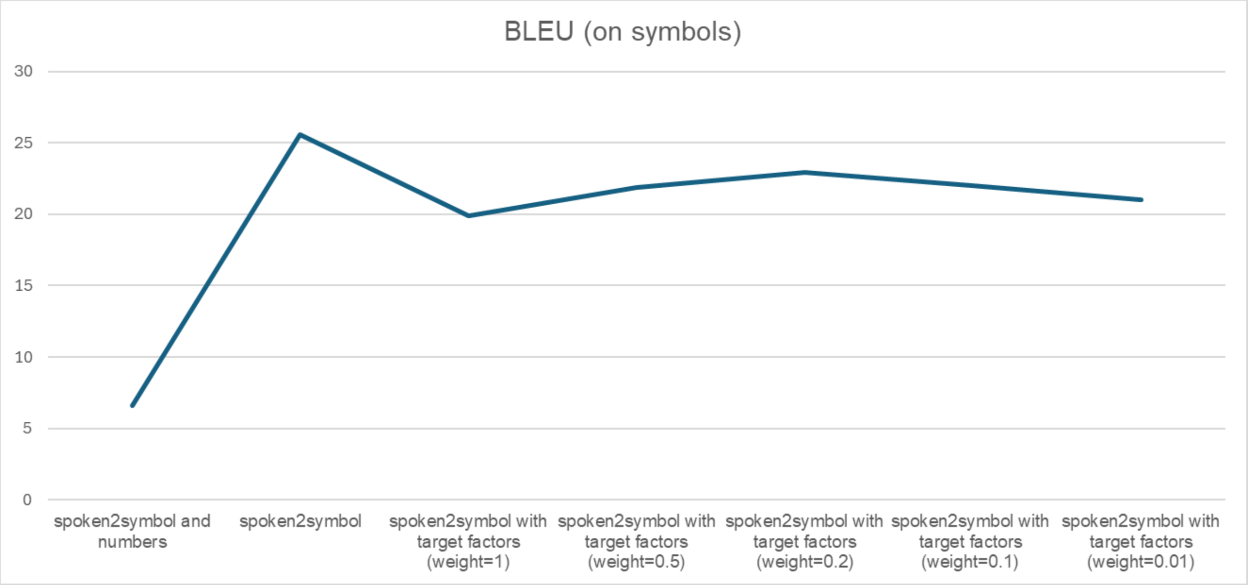


Figure 17. *Text2Signwriting2Sign Models' BLEU-4 Score.*

Based on the BLEU result table for models that convert from spoken language into signs (text2sign), we can draw some important comments on the overall performance of the models.

The "Spoken2Symbol" model achieves the highest BLEU point of 25.6, showing that it works effectively in the conversion from the text to the symbol without additional elements. In contrast, the "Spoken2Symbol and Numbers" model only reaches BLEU as 6.6, which proves that adding digital components has significantly reduced the accuracy of the model. The reason may be due to the complexity of this factor that makes it difficult for the model to accurately capture the symbols.

When considering the effects of the weight of additional factors, we see that the BLEU performance varies depending on the applicable weight. Specifically, with the number 1, the BLEU score reached 19.9, lower than the model only "Spoken2Symbol" (25.6), showing that the use of additional factors with maximum weight does not bring high efficiency. When the weight decreased to 0.5, the BLEU point increased to 21.9, and reached the highest when the weight was 0.2, with the BLEU point of 22.9. This shows that the use of additional factors with average weight can improve accuracy, but should not exceed certain weight. When the weight continues to decrease to 0.1, the BLEU point remains at 22, but when it drops to 0.01, the BLEU point decreases to 21. This indicates that decreasing weight to too low can reduce the effectiveness of additional factors.

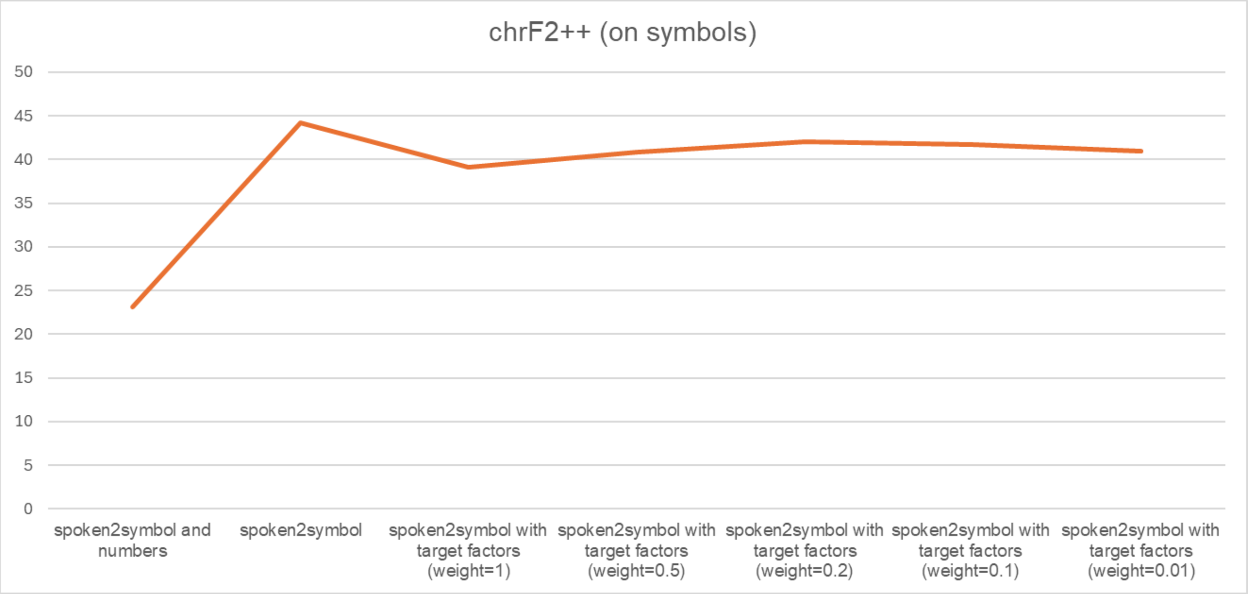


Figure 18. *Text2Signwriting2Sign Models' CHRF Score.*

The overall efficiency of the models shows that the “Spoken2Symbol” model achieves the highest CHRF2 ++ score of 44.2, showing the best performance in switching from text to symbols without additional elements. This result is consistent with the previous BLEU index, proving that the basic model operates effectively without further adjustment. In contrast, the “Spoken2Symbol and Numbers” model achieves the lowest CHRF2 ++ score of 23.1, showing significantly less performance when there is an additional digital component, possibly due to the increased complexity from this factor.

When analyzing the effects of the weight of additional elements, we realize that the CHRF2 ++ score changes the same as the BLEU index, but it seems more stable. With the number 1, the score of CHRF2 ++ is 39.1, lower than the pure “Spoken2Symbol” model (44.2), showing that the maximum use of the additional weight is not the optimal choice. When the weight decreased to 0.5, the CHRF2 ++ point increased to 40.8, and continued to peak at 0.2 with a point 42, showing that the average weight helps improve performance. Even if the weight is reduced to 0.1 and 0.01, the CHRF2 ++ score is still relatively high, respectively 41.7 and 40.9, respectively, showing that the weight reduction does not affect too much on this index while it may reduce noise.

When comparing between BLEU and CHRF2 ++ indicators, both showed the “Spoken2Symbit” model that works best without additional factors. The average weight (from 0.1 to 0.5) gives additional elements to help the model achieve better results than other weights (1) or too low (0.01).

5.2.2. Sign2Signwritng2Text

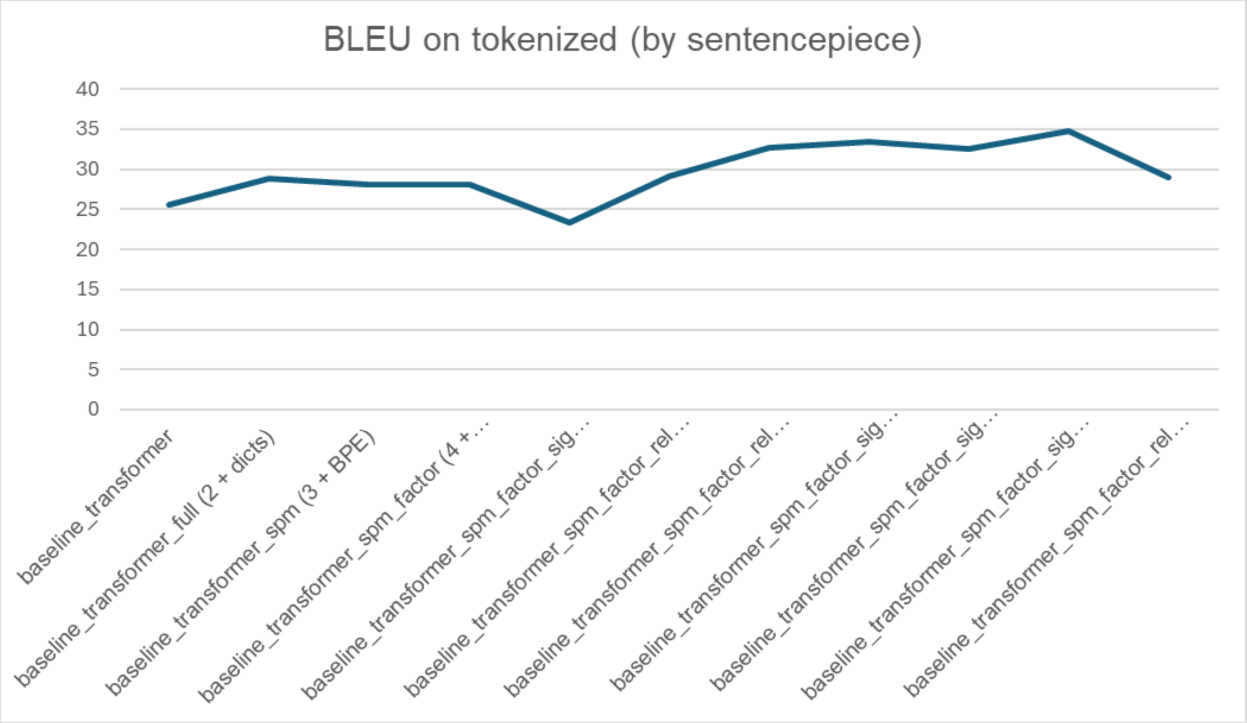


Figure 19. *Sign2Signwriting2Text Models' BLEU-4 Score.*

The overall performance of the models shows the basic model “Baseline_transformer” achieved BLEU 25.6, which is the starting point to compare with the optimized models. Adjustable models often have higher BLEU points, showing the effectiveness of refining in improving the accuracy of the model.

When considering the effects of various technical improvements, the “full” (2 + dicts) configuration with the full dictionary helps to increase the BLEU to 28.8, showing the use of the dictionary is beneficial for the results pandemic. Models “Baseline_Transformer_Spm” and “Baseline_Transformer_Spm_Factor (4 + X and Y)” Achieving BLEU points are 28.1 and 29.1, showing that the tokenization with SentencePiece and the addition of additional elements for X and Y coordinates can be Help the model better grasp complex symbols.

However, when adding FSW (Factored Signwriting) as an additional element in the “Factor with FSW (5 + Factored FSW Graphemes)”, the BLEU point is reduced to 23.4, this may be due to this supplement to this factor. Increase complexity without improving model efficiency. Meanwhile, the use of relative X and Y elements in “Factor with relative X and Y (5 + Relative x and Y)” helps to achieve BLEU 29.1, higher than absolute factors, shows that the relative factor can help the model better understand the context.

The adjustment of Dropout with “tieD_softmax” in the model (7) has helped to increase the BLEU to 32.7, showing that the application of a reasonable Regularization improves accuracy. Using FSW as an additional element in the model (8) to achieve the BLEU 33.5 point, higher than using both FSW for both source and additional factors, showing that only FSW is used for additional factors is how to be how to More optimal. When not using Lowercasing in the model (9), the BLEU point reaches 32.6, showing that keeping the capital letters can increase the accuracy of some specific words in the context.

Reducing BPE Vocabulary size from 2000 to 1000 of the model (10) has led to the BLEU point of 34.8, which is one of the highest results, showing that reducing the size of the BPE vocabulary can help the model be more optimal, it may be due to noise from unnecessary tokens. Although adding “de dict” in the model (10) only achieved the BLEU 29 point, showing a slight decrease, this shows that this dictionary may not be optimal in this case.

Conclusion, models that have adjusted Dropout, tieD_softmax, and FSW element when used appropriately have significantly improved performance compared to the basic model. Configuration “Baseline_transformer_spm_Factor_sign+ (10)” with a smaller vocabulary size of BLEU 34.8, is the highest result, showing that the model can be optimized with a smaller vocabulary. Factors such as keeping capital letters and using relative coordinates for symbols are small but useful improvements. The best model for the translation of sign2text in this case is “baseline_transformer_spm_factor_sign+ (10)” with a smaller vocabulary, because it reaches the highest BLEU point.

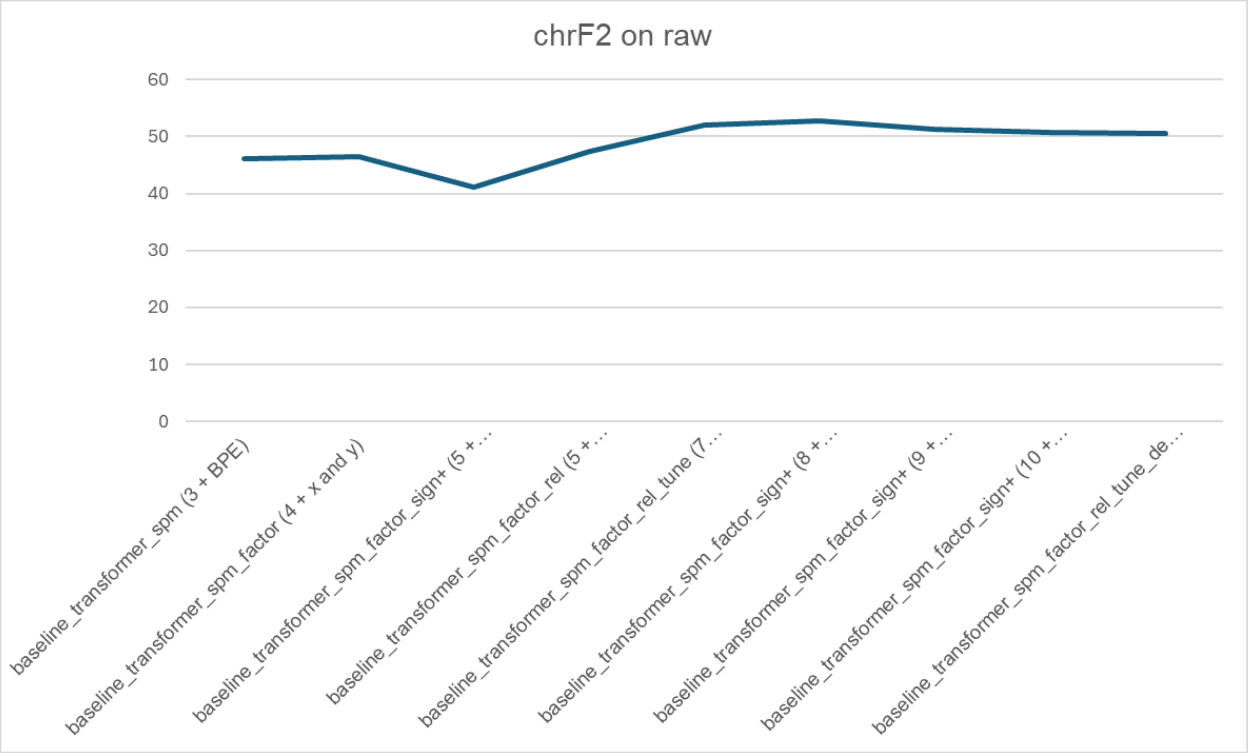


Figure 20. *Sign2Signwriting2Text Models' CHRF Score.*

The baseline model (Baseline_Transformer) reaches the lowest CHRF2 point, showing that adding additional processing techniques can improve the performance of the model. The addition of BPE (Baseline_Transformer_SPM) has improved its performance, proving that coding-code can enhance the capability of the model's context. Using factors to model graphemes (baseline_transformer_spm_factor) often brings better performance, this shows that exploiting information about graphemes can help the translation model more accurately. The method of combining elements (sign+ and reling) often achieved better results than using each individual method, this proves that combining information from many sources can improve the efficiency of tissue effectiveness. image. Finally, improving the Fine-Tune (reliable) can raise the CHRF2 score, showing that the model optimization for specific data sets is more effective.

5.2.3. Final model

Through the results, the authors found: (1) For the model for the "Text2Signwriting2Sign" stage, the "spoken2symbol" model brings the highest BLEU-4 and CHRF results, with BLEU-4 being 25.6 and CHRF being 44.2. (2) For the model for the “Sign2Signwriting2Text” stage, the model “baseline_transformer_spm_factor_sign+ (10 + smaller bpe vocab 2000 -> 1000)” gives the highest BLEU-4 result, 34.8, and CHRF is 50.8 although not high but there is not much difference from the model with the highest CHRF2 (52.7), so we decided to choose this model.

Below is a table comparing the BLEU-4 and CHRF indices of all 3 of our approaches in this article, in which for approach 3 we only choose 2 models that bring the best results. For Approach 1, we do not have results for BLEU-4 and CHRF because the training process is interrupted at the Gloss2Text step due to not having enough data to translate gloss to text as we presented in section 3.2.

Table 3. *Summary of 3 Sign Language Translation Approaches and Performance Metrics.*

No.	Model	Type	Data	BLEU-4	CHRF
Approach 1	Sign2Gloss2Text	Translate Video Sign Language to Text	MS-ASL	-	-
Approach 2	basic Transformer seq2seq endocder-decoder (baseline)	Translate Video Sign Language to Text	How2Sign	8.43	9.6
Approach 3	spoken2symbol	Translate Video Text to Sign Language	SignBank+	**25.6**	**44.2**

	baseline_transformer_spm_factor_sign+ (10 + smaller bpe vocab 2000 -> 1000)	Translate Video Sign Language to Text	SignBank+	**34.8**	**50.8**

For the translation process “Text2Signwriting2Sign” and “Sign2Signwriting2Text”, here is an example of calling the API and getting the following results:

Table 4. *Example of Final Model API testing*

	Requests	**Returns**

Text2Signwriting2Sign	*POST “http://127.0.0.1:3030/api/translate/spoken2sign*
	{
	"country_code": "us",
	"language_code": "en",
	"text": "hello",
	"translation_type": "dict"
	}
		{
		"country_code": "us",
		"direction": "spoken2sign",
		"language_code": "en",
		"n_best": "3",
		"text": "hello",
		"translation_type": "dict",
		"translations": [
		"M518x518S30007482x483S15a11485x482S26500515x461",
		"M518x518S14c20481x453S27106503x483",
		"M518x518S30a00482x483S33e00482x483"
		]
		}

Sign2Signwriting2Text	*POST “http://127.0.0.1:3030/api/translate/sign2spoken* { "country_code": "us", "language_code": "en", "text": "M528x518S15a07472x487S1f010490x503S26507515x483 M517x515S1e020494x485S21801482x492 S38800464x496 L532x594S2e74c475x569S30d00482x482S2e700517x555S14220500x520S14228473x533 L518x571S2ff00482x482S30d00482x482S22a03480x536S15a51461x548S15a00494x508 L536x527S10e41465x485S2892a507x494S10e4a474x512S2892a506x472 S38700463x496	{ "country_code": "us", "direction": "sign2spoken", "language_code": "en", "n_best": 5, "text": "M528x518S15a07472x487S1f010490x503S26507515x483 M517x515S1e020494x485S21801482x492 S38800464x496 L532x594S2e74c475x569S30d00482x482S2e700517x555S14220500x520S14228473x533 L518x571S2ff00482x482S30d00482x482S22a03480x536S15a51461x548S15a00494x508 L536x527S10e41465x485S2892a507x494S10e4a474x512S2892a506x472 S38700463x496 L529x568S15a37477x442S18250488x453S28802507x431S15a37481x536S10051492x547S26901470x522 L526x599S15a50474x528S15a31479x576S24103494x545S30122482x476 S38900464x493 M530x679S1001a490x568S10041487x541S2e718466x654S22f04492x527S14220498x607S2e73c514x

	L529x568S15a37477x442S18250488x453S28802507x431S15a37481x536S10051492x547S26901470x522 L526x599S15a50474x528S15a31479x576S24103494x545S30122482x476 S38900464x493 M530x679S1001a490x568S10041487x541S2e718466x654S22f04492x527S14220498x607S2e73c514x642S14228467x618S30a00482x482 M547x531S1853f473x504S18537513x497S2b711474x479S2b700508x469S2b700528x480S2b711452x492 S38800464x496", "translation_type": "sent", "n_best": 5 }	642S14228467x618S30a00482x482 M547x531S1853f473x504S18537513x497S2b711474x479S2b700508x469S2b700528x480S2b711452x492 S38800464x496", "translation_type": "sent", "translations": ["Verse 21. The wicked borrow and again, and the godly gives them.", "Verse 21. The wicked borrow and never repay, but the godly gives them.", "Verse 21. The wicked borrow and never repay, but the godly gives it.", "Verse 21. The wicked borrow and never repay, but the godly gives them.", "Verse 21. The wicked borrow, never repay, but the godly gives them."] }

6. Conclusion and Future Work

6.1. Conclusion

The current project has achieved significant results in developing a sign language translation system. It enables the conversion of text into sign language and vice versa from sign language to text for three countries: America, France, and Germany. Besides, the project is extended for more sign languages, though the extension is incomplete.

The project has been able to demonstrate that it is very possible to come up with a communication support tool for the deaf within the community. This would not only avail users with easy access to information but also promote their capabilities of communicating effectively with persons who do not use sign language. Also, the project has extended the ability to translate not just American Sign Language but some other countries' sign languages, thus improving communication for the hearing impaired in various nations.

Another limitation of this project in translating spoken language to sign language is that sign language grammar has not been perfectly portrayed yet. Grammatical rules of the signs differ from the ones used in spoken language, and the translation system has not yet been perfected to reflect these structures in results of translations. Because of this, some of the translations could be tricky to understand and sometimes look unnatural.

6.2. Future Work

In the future, one of the main aims is to complete the system, as there are still limitations to address. Additionally, improving the service by adding more supported languages is a key goal. The system will be further updated to support more languages and provide an accuracy that is at par with the three currently supported languages. This will enhance not only globalization but also user satisfaction across different countries.

Other key enhancements shall be put in place with a focus on overcoming what can be termed the worst limitation of the existing system is the grammatical accuracy that has failed to fully translate spoken languages into sign language. Therefore, the future enhancement of the system shall be focused on developing a more accurate translation model, which shall adhere to the grammatical rules of sign language. The translated text shall always provide proper and natural meanings for the hearing-impaired person, making communication easier.

References

[1] Zifan Jiang, Amit Moryossef, Mathias Müller, and Sarah Ebling. (2023). *Machine Translation between Spoken Languages and Signed Languages Represented in SignWriting*. In Findings of the Association for Computational Linguistics: EACL 2023, pages 1706–1724, Dubrovnik, Croatia. Association for Computational Linguistics.

[2] Amit Moryossef. *SignWriting Hands Classification v2*. Google Colab. Retrieved [16-10-2024] from https://colab.research.google.com/drive/1xDXYyQ_U__jMzjRYKG2L55D43hJ11u-l

[3] J22Melody. (n.d.). *SignWriting Translation*. GitHub. Retrieved [10-10-2024], from https://github.com/J22Melody/signwriting-translation

[4] Camgoz, N. C., Koller, O., Hadfield, S., & Bowden, R. (2018). *Neural Sign Language Translation.* CVPR, 7784-7793.

[5] Yin, S., Xia, Z., Chen, X., Zhou, H., & He, S. (2020). *Sign Language Translation with Transformer*. MM '20, 1778–1786.

[6] Orbay, E., & Akarun, L. (2020). *Neural Sign Language Translation by Learning Tokenization.* arXiv:2004.03519.

[7] Camgoz, N. C., Hadfield, S., Koller, O., & Bowden, R. (2020). *Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation*. CVPR, 10023-10033.

[8] Saunders, B., Camgoz, N. C., & Bowden, R. (2020). *Progressive Transformers for End-to-End Sign Language Production.* ECCV, 687–705.

[9] Pu, J., Zhou, P., Wang, F., & Xu, W. (2019). *Boosting Continuous Sign Language Recognition via Cross Modality Augmentation.* CVPR, 11509-11518.

[10] Amit Moryosse, Zifan Jiang. (2024). *SignBank+: Preparing a Multilingual Sign Language Dataset for Machine Translation Using Large Language Models.*

[11] Amy LC. (2018). *Glossing in ASL. What is it? Eight examples.* SlideShare. Retrieved [20-8-2024].

[12] Padden, C., & Humphries, T. (2005). *Inside Deaf Culture*. Harvard University Press.

[13] Hanke, T. (2004). HamNoSys - Representing sign language data in language resources and language processing contexts. *4th International Conference on Language Resources and Evaluation (LREC)*.

[14] Sutton-Spence, R., & Boyes Braem, P. (2012). *Hand and Mind: What Gestures Reveal about Thought*. University of Chicago Press.

[15] Neha Keshari, Anil Kumar Srikantham, Gerardo Gomez Martinez. *Real-time ASL to English text translation*. Standford CS230.

[16] Tarrés, L., Gállego, G. I., Duarte, A., Torres, J., & Giró-i-Nieto, X. (2023). *Sign language translation from instructional videos.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 5625-5635).