Crafting Sign Language From Speech

Nya Haseley-Ayende, Javier Fernandez Garcia, Halleluiah Girum, Bahar Birsel, Amaris Grondin

Link to Github: https://github.com/Javierfg02/Cutlass-Fish

Introduction

Sign language translation holds profound significance in fostering inclusivity and accessibility for individuals with hearing impairments, a population that encompasses approximately 466 million people worldwide, according to the World Health Organization.[1] In the United States alone, around 15% of adults aged 18 and over report some trouble hearing, as reported by the National Institute on Deafness and Other Communication Disorders (NIDCD)[2]. Within this demographic, sign language serves as the primary mode of communication for approximately 250,000 to 500,000 individuals in the U.S., as per the National Association of the Deaf (NAD).[3] Sign language translation promotes equity by ensuring that information and resources are accessible to all, regardless of auditory ability. It enables individuals with hearing impairments to participate more actively in society, promoting their rights to equal opportunities and inclusion. By developing effective sign language translation tools, we empower this community to engage more fully in various aspects of life, including education, employment, social interactions, and accessing essential services.

Motivated by this mission to foster inclusivity and accessibility for individuals with hearing impairments, our group aims to bridge the communication gap between people who are deaf or hard of hearing and those who are not by creating a tool that translates sentences of speech to sequences of sign language. There has been significant research into sign language translation (SLT) models that enable one-way communication between those who are deaf or hard of hearing and others by converting sign language into speech. However, to enable two-way communication between individuals with and without hearing impairments, we must also translate speech into sign language which we do through a new technique in the field known as sign language production (SLP). Our project aims to bridge this communication gap by

[1] World Health Organization. "Deafness and hearing loss." Accessed May 6, 2024. https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss.

[2] National Institute on Deafness and Other Communication Disorders. "Quick Statistics About Hearing." Accessed May 6, 2024. https://www.nidcd.nih.gov/health/statistics/quick-statistics-hearing.

[3] National Association of the Deaf. "About Deafness."Accessed May 6, 2024. https://www.nad.org/about-deafness/.

developing a tool that translates sequences of English speech into sequences of American Sign Language.

1.1 Related Work

To bridge the communication gap between individuals with hearing impairments and those without, we implemented a similar approach as the one presented in the research paper "Progressive Transformers for End-to-End Sign Language Production" by Saunders et al.[4] The paper outlines a novel approach aimed at achieving bidirectional translation between sign language and spoken language. Saunders et al.'s work stands out among previous studies due to its innovative use of the 'Progressive Transformer' architecture. This architecture is specifically designed to accommodate the continuous nature of sign language sequences, addressing a crucial aspect often overlooked in earlier research efforts.

In another research paper, Zelinka et al. explored text-to-video sign language synthesis by developing a neural network-based translator capable of converting text to synthesized skeletal pose. Their study utilized freely available data from daily TV broadcasts, omitting video segmentation. The primary objectives were twofold: first, to extract high-quality skeletal models from videos; second, to establish a fully trainable end-to-end sign language synthesis system without explicit translation. For the former task, the researchers employed Open-Pose, a neural network-based method for skeleton extraction. Regarding the latter task, they implemented a cutting-edge seq2seq translator utilizing an RNN-based encoder-decoder technique with LSTM, incorporating an attention mechanism. Notably, they replaced CNNs and the tokenization layer with a word embedding layer and replaced the softmax activation function in the output with a linear function. Evaluation of their findings was conducted using Mean Squared Error (MSE) with Dynamic Time Warping (DTW), an algorithm assessing similarity between temporal sequences with variable speed.[5]

Our approach involves utilizing Tensorflow rather than Pytorch, as well as an entirely different dataset than that of Saunders et al’s work, due to our motivation for working closely with American Sign Language (ASL) translation rather than German

[4] Saunders, Ben, Necati Cihan Camgoz, and Richard Bowden. "Progressive Transformers for End-to-End Sign Language Production." In Proceedings of the European Conference on Computer Vision (ECCV), 2020. Accessed May 6, 2024. https://arxiv.org/pdf/2004.14874.pdf

[5] Zelinka, Miroslav, et al. "Neural Sign Language Synthesis: Words Are Our Glosses." Proceedings of the Winter Conference on Applications of Computer Vision (WACV), 2020. Accessed May 6, 2024. https://openaccess.thecvf.com/content_WACV_2020/html/Zelinka_Neural_Sign_Language_Synthesis_Words_Are_Our_Glosses_WACV_2020_paper.html.

Sign Language (GSL) translation. Moreover, the dataset used by Saunders et al was missing data for facial movements during signing, which is a key component that our dataset includes. However, our objective aligns with the core aim of the study: to enhance inclusivity by facilitating seamless translation in both directions.

1.2 Code Repository

Our codebase is publicly accessible on GitHub here.

Methodology

2.1 Dataset

In our project, we employed the How2Sign dataset, a comprehensive resource comprising more than 80 hours of video content translating English sentences into American Sign Language (ASL). Specifically, our analysis focuses on the pre-extracted body-face-hands key points derived from the translated videos, providing detailed descriptions of the signers' postures at each frame.

2.2 Preprocessing and the Model

The data underwent preprocessing, resulting in a source field of tokenized English sentences and a target field consisting of their corresponding key-point values describing the pose translation, as well as the creation of a vocabulary from the tokenized English sentences. In particular, each of the tokenized English sentences was mapped to its index in our vocabulary dictionary. This mapping generated two tensors: one representing padded sequences with zeros and another representing the original lengths of the sentences. Our model architecture is based on the Progressive Transformer, as proposed by Saunders et al. This architecture consists of two main components:

- **Symbolic encoder:** Converts spoken language (text) into a symbolic embedding representation using multi-head attention layers. As with a traditional transformer, this should generate context-aware embeddings that capture the semantics of the source English sentences.
- **Progressive decoder:** Translates the symbolic embeddings directly into continuous sign pose sequences. This is achieved by utilizing a counter decoding technique.
    - Counter decoding technique: Each sign pose frame is embedded along with a progress counter value ranging from 0 to 1 to track the progress of sentence generation. The model then predicts both the next sign pose frame and the progress counter at each step, stopping at counter = 1.

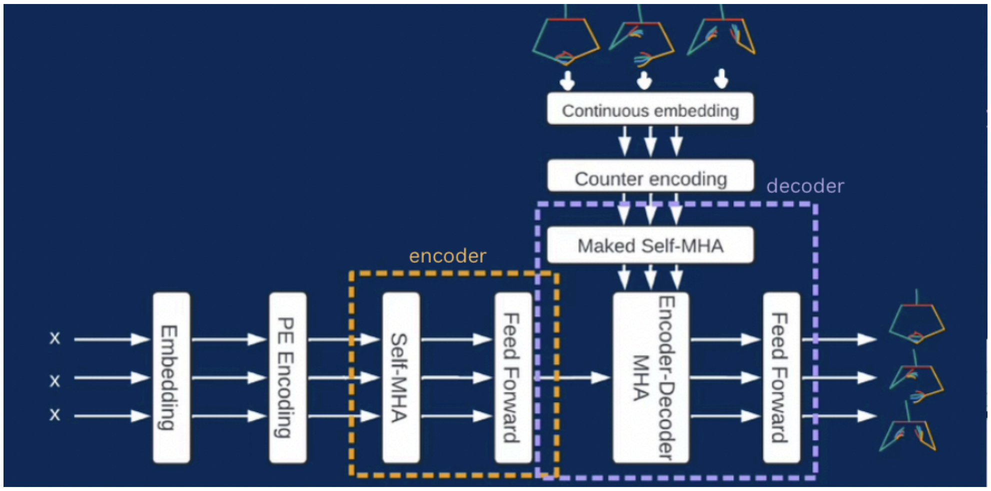

Figure 1: A visual representation of the model’s architecture

2.3 Training and optimization

Due to the lack of video output from our model, we have experimented with both our model and the author’s model.

Our model was trained for 250 epochs converging to a loss of 0.00476 after ~65 epochs with a dynamic learning rate from 0.001 to 0.0002. Losses were computed using Mean Squared Error (MSE) between the predicted and actual sign poses, and unfortunately we could not quote dynamic time warping results from our model due to the lack of video output.

The author’s model was trained for 500 epochs using an Adam optimizer with a dynamic learning rate ranging from 0.001 to 0.0002. Losses were computed using Mean Squared Error (MSE) between the predicted and actual sign poses, with additional evaluation metrics including dynamic time warping (DTW) which assesses alignment of the produced sign pose sequences with the ground truth.

Results

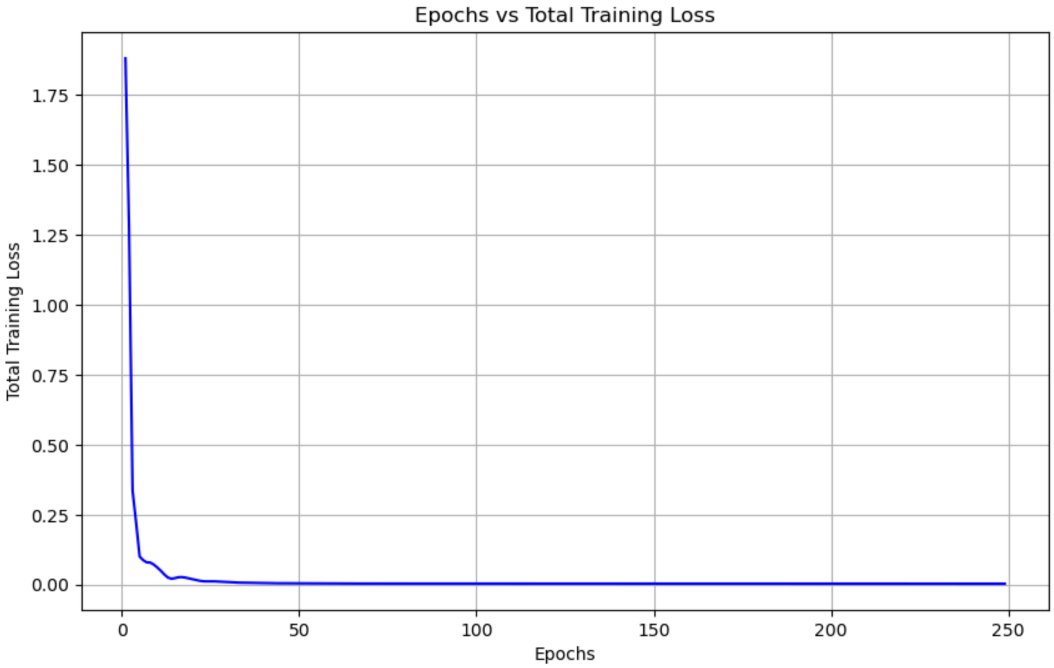

Figure 2: Training loss vs epochs with our model

The figure above shows a plot of our training loss against the number of epochs for which the model was trained. Overall, it can be concluded that our training loss converges to 0.00476 at epoch ~65.

Comparing our results with the paper's results which approaches 0.007 across 500 epochs, we can observe that our loss is about the same as their loss, though notably, we were only able to compare the training loss and not the testing losses for which our results are less likely to agree.

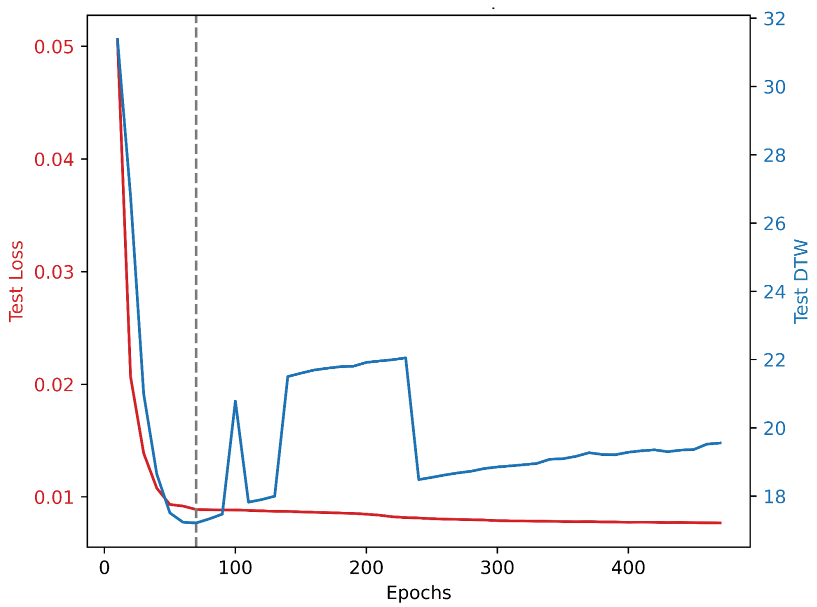

Figure 3: Test loss and DTW vs Epochs for run with author's model

The graph above shows the test loss for each epoch for the 500 epochs for which we trained the author's model. Similarly to what we observed with our model, the loss converges after 70 epochs, at which point the DTW score worsens. A possible explanation for this is that the model learns to minimize the MSE loss by twirling the fingers in the space where they are usually found while signing, without improving sequence similarity between the videos. In other words, the model is producing sign language sequences that minimize the loss but are not temporally aligned well with the ground truth, leading to an increase in DTW. The model is essentially overfitting.

An example of the aforementioned phenomenon can be seen in the output videos from the author's model:

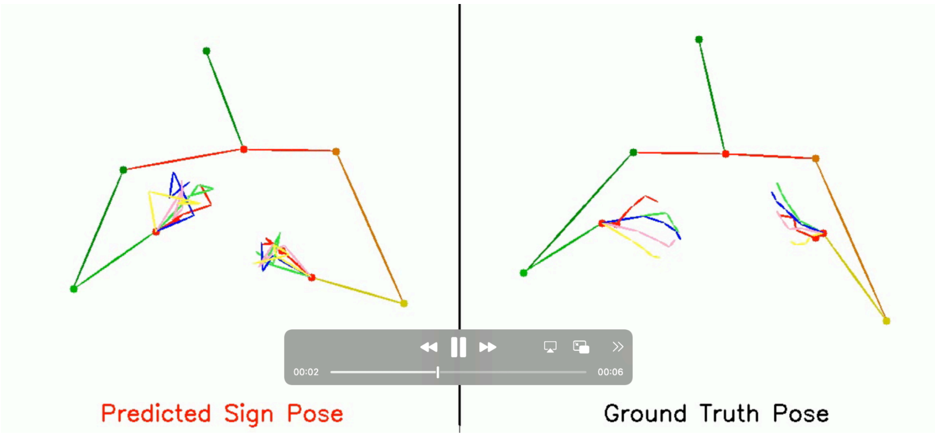

Figure 4: Model's predicted pose (left) vs ground truth (right)

Here, it is likely that the small loss is due to the model minimizing the MSE error by appropriately placing the shoulders and arms, but simply wiggling the fingers around in the place where they are most likely to be without properly aligning the predicted and ground truth poses, and thus not correctly producing interpretable sign language sequences.

Challenges

The most challenging aspect of the project involved our data preprocessing. Our new dataset comprised both sign skeletons and text. This means that we needed to link each skeleton and its corresponding text via an ID within a dictionary. Moreover, our dataset diverged from the one utilized in the referenced paper; it segmented the skeleton into distinct components such as the face, right hand, left hand, and body. Consequently, we merged these segmented components into a unified skeleton while minimizing information loss resulting from dimensionality reduction. Ensuring compatibility with our existing codebase further compounded this challenge.

Furthermore, we faced significant difficulties building the transformer model. Tensorflow implementations are not viable for continuous SLP with a counter decoder, thus we had to build a modified multi-headed attention transformer with counter decoding (the progressive transformer) from scratch. In this variant, the decoder has to predict both a sequence of skeleton outputs and the counter value for each skeleton, introducing additional complexity to the model architecture.

Lastly, navigating and understanding the extensive codebase authored by the researchers of our chosen paper was also very challenging. The paper's code contained over 20 distinct classes of code using many techniques previously unfamiliar to us or involving complicated translations from Pytorch to Tensorflow, such as the counter decoding technique and various data augmentation methods, as well as GPU optimization strategies. We learnt a lot as we progressed through the code in understanding and integrating these advanced techniques into our project.

Reflection

How do you feel your project ultimately turned out? How did you do relative to your base/target/stretch goals?

While we are content with the effort we put into the project and the huge amounts that we have learnt from it such as understanding deep learning research papers and architectures, scouring and preprocessing data and navigating real-world projects as larger team, we are disappointed that our model did not work as we intended it to as we were able to train the model but encountered bugs when trying to get a video output from it. Going into this project, we knew that our task, being an extremely novel area in deep learning research, was challenging, but we were hoping that we would be able to obtain concrete results from our model.

We did achieve our base goal which was to train a model from speech (english sentences) to sign language with relatively low loss. However, due to time constraints, we were not able to complete our reach and target goals of being able to translate the sign back into speech in order to compare with the original text.

Did your model work out the way you expected it to?

Our model did not work out the way we expected it to. We believe that we are close to having our model working as expected, since the model trains with relatively low loss but there are bugs when we try to output the videos using the validation dataset. Some key issues we solved include adding padding in order to handle sequences of varying lengths, as well as shape compatibility issues which arose during masking in the decoder.

From the author’s model, our expectation was for the model to work with reasonable enough accuracy that in the video's output some sign language signs would be correctly translated. For this to occur, the dynamic time warping score must be relatively low (< 8) while the current lowest score is 16.

However, given the extensive debugging required to even get our model to train, we do believe we have arrived at a good place. In the future, we would like to get the video production to work and after that we can use some of the regularization techniques mentioned in the paper such as adding gaussian noise in order to achieve understandable sign language.

How did your approach change over time? What kind of pivots did you make, if any? Would you have done differently if you could do your project over again?

Our approach became more strategic as the project progressed. Initially, we planned to replicate Saunders et al. 's work closely but realized that a direct replication wasn't feasible due to differences in dataset and software frameworks. One of our initial pivots was the decision to use the Progressive Transformer for Text-to-Pose (T2P) directly, without gloss intermediaries. Our strategy also pivoted towards the midpoint of the project, after having processed the data, as we realized more and more that it was not feasible to mimic the work done by Saunders et al, and we focused heavily on understanding their work deeply so that we could adapt it to meet our data format and language requirements. If we could re-do the project, we would focus more on incremental model building and testing, ensuring compatibility between the dataset and model layers. We believe much of our issues stemmed from implementations of classes and functions that were not rigorously tested before being used in other areas of the project, at which point we had buried logical issues layers down the code.

What do you think you can further improve on if you had more time?

If we had more time, we would have liked to take this project slower. With the extensive code base that we had to navigate and thoroughly understand, and the complexity of the architecture we had to implement, we believe that a slower approach where we could have rigorously tested our implementations before moving on to the next step would have avoided many of the countless bugs we encountered along the way.

Also, our loss currently relies on the difference of the position of all body parts including the body, arms and fingers of our model's output and the groundtruth. If we had more time, it might have been beneficial to weight the loss to take into account the position of the fingers more than the rest of the body because they are the most important part of conveying the meaning of the sentence.

What are your biggest takeaways from this project/what did you learn?

The biggest takeaways from this project include a greater understanding of the world of deep learning research. Having read and deeply understood multiple deep learning research papers was as intriguing as it was educational. We dove into the complexities of improving current state-of-the-art architectures which we all found as challenging as it was rewarding.

Along with this, we also learned the importance of data preprocessing, recognizing that a well-defined preprocessing pipeline is crucial, especially when handling multi-modal data like sign language, and we developed the skills to do this effectively. Additionally, we developed skills in model debugging and optimization, as the large amounts of data and complex models that we were dealing with would not even begin to function without optimizations.

Lastly, this project was one of the few opportunities during a computer science degree where we get to work together as a moderately large group, just like we will have to very soon when we graduate. While in the beginning we struggled to coordinate and divide tasks effectively, which led to many dreaded github conflicts, soon enough we found ourselves fluidly building on top of each other’s work.

smartView → data to microsoft sheets

Different endowments → 3 million dolllars