Gloss2Text: Sign Language Gloss translation using LLMs and Semantically Aware Label Smoothing

Pooya Fayyazsanavi[1], Antonios Anastasopoulos[1,2], Jana Košecká[1]
[1]George Mason University, USA
[2]Archimedes Research Unit, Athena RC, Greece
{pfayyazs,antonis,kosecka}@gmu.edu

Abstract

Sign language translation from video to spoken text presents unique challenges owing to the distinct grammar, expression nuances, and high variation of visual appearance across different speakers and contexts. Gloss annotations serve as an intermediary to guide the translation process. In our work, we focus on *Gloss2Text* translation stage and propose several advances by leveraging pre-trained large language models (LLMs), data augmentation, and a novel label-smoothing loss function that exploits gloss translation ambiguities, significantly improving the performance of state-of-the-art approaches. Through extensive experiments and ablation studies on the PHOENIX Weather 2014T dataset, our approach surpasses state-of-the-art performance in *Gloss2Text* translation, indicating its efficacy in addressing sign language translation and suggesting promising avenues for future research and development.[1]

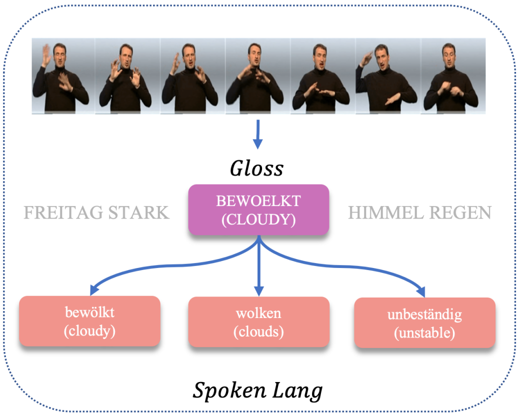


Figure 1: An example of ambiguity in sign language is demonstrated by the gloss "BEWOELKT (CLOUDY)," which is represented in multiple translations within the dataset. As shown, ambiguity may share the same meaning but differ in form, such as "wolken (cloudy)," or where the gloss represents the concept meaning, such as "unbeständig (unstable)."

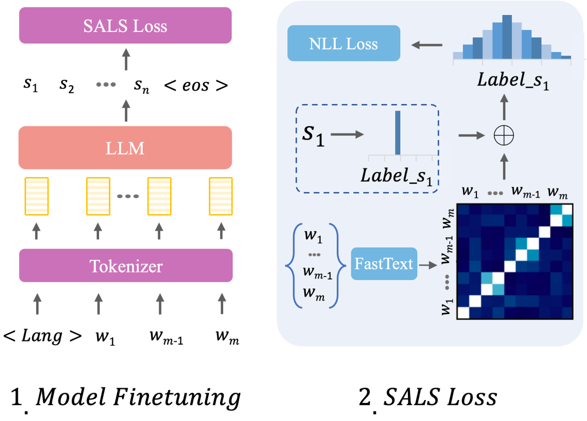
1 Introduction

Sign language translation from video to spoken text often involves two phases: *Sign2Gloss* and *Gloss2Text*. In *Sign2Gloss* phase, the gloss annotations, are predicted from input videos as shown in the top part of Figure 1, establishing a link between visual expressions and corresponding meanings. The subsequent *Gloss2Text* phase, translates these gloss annotations into spoken language. While gloss annotations have strong limitations as linguistic representations Angelova et al. (2022), the emergence of pre-trained large language models, word embeddings, and advances in Machine Translation open up new possibilities for improvements in *Gloss2Text* translation task. In our work, we propose to leverage Large Language Models (LLMs) pre-trained on expansive and diverse corpora along with novel sign language specific label smoothing loss and data augmentation techniques to improve *Gloss2Text* phase of sign language translation task. Our contributions are the following:

- Development of tailored data augmentation techniques for *Gloss2Text* translation, including paraphrasing to enhance spoken aspects by proxy language translation, and back-translation for gloss augmentation.
- Novel label-smoothing loss function optimized for gloss translation specific ambiguities, reducing penalties for incorrect predictions that are similar to the target translation.
- State-of-the-art performance in *Gloss2Text* translation, surpassing existing benchmarks on the PHOENIX Weather 2014T dataset and detailed ablation study of different components of our approach.

[1] Code can be found at the following GitHub repository: https://github.com/pooyafayyaz/Gloss2Text

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 16162–16171
November 12-16, 2024 ©2024 Association for Computational Linguistics


SALS Loss
s_1 s_2 ••• s_n < eos >
LLM
Tokenizer
< Lang > w_1 $w_{m\text{-}1}$ w_m
NLL Loss
Label_s_1
s_1
Label_s_1
w_1 ••• $w_{m\text{-}1}$ w_m
w_1 ••• $w_{m\text{-}1}$ w_m
FastText
w_1 ••• $w_{m\text{-}1}$ w_m

1. *Model Finetuning* 2. *SALS Loss*

Figure 2: The proposed architecture for ***Gloss2Text*** translation. Initially, the similarity of each word to others is compared. During training with label smoothing, depicted on the left side, the model aims to identify the most similar words to the target word and assign heightened labels to those words.

2 Approach

The goal of our gloss translation system is to convert a series of gloss annotations $G = g_1, g_2, \ldots, g_T$ into a spoken word sequence $T = t_1, t_2, \ldots, t_L$. given n pairs where each pair can have different input and output lengths. Our approach involves fine-tuning large language models tailored specifically for our task.

To effectively train a typical Neural Machine Translation (NMT) model, a corpus of around 1 million parallel samples is often required Sennrich and Zhang (2019). However, the existing sign language datasets are of several orders of magnitude smaller. For instance, the PHOENIX-2014T German sign language dataset Camgoz et al. (2018), the most widely benchmark for continuous sign language, has only 8,257 gloss-text pairs.

One group of approaches Chen et al. (2022a,b); Zhou et al. (2023) has concentrated on fine-tuning LLMs for the sign language *Gloss2txt* translation task without data augmentation. A series of studies Ye et al. (2023); Zhu et al. (2023); Angelova et al. (2022); Zhang and Duh (2021) have investigated limited data augmentation techniques to address the challenge of data scarcity.

In our experiments, we propose several strategies for fine-tuning NLLB-200 Costa-jussà et al. (2022) model. Additional experiments with alternative models including MT5 Xue et al. (2020), mBart Liu et al. (2020) can be found in Appendix B. These models share multilingual characteristics, enabling them to handle diverse language pairs efficiently. However, they differ in the datasets they are pre-trained on, the specific architectures they employ, and their respective training objectives, which affect their performance in sign language translation tasks.

2.1 Data Augmentation

To improve the robustness of the baseline translation approach, we explore two distinct data augmentation techniques.

Paraphrasing translates the original target sentence into a proxy language (English) and then back to the original (German). This cycle introduces linguistic diversity on the target side while ideally preserving the original meaning in gloss annotations, exposing our model to a broader spectrum of linguistic variations.

Back Translation involves training a reverse translation model with the spoken data as input, producing the corresponding gloss sequence. If the model's generated gloss sequence differs from the original one, we iterate over sentences and incorporate this new gloss sequence as a silver label alongside the translation pair for our primary training process.

2.2 Semantically Aware Label Smoothing

In the conventional label smoothing approach Szegedy et al. (2016); Müller et al. (2019) one replaces one-hot encoded label vector $\mathbf{y}_{hot}$ with a mixture of $\mathbf{y}_{hot}$ and the uniform distribution $\boldsymbol{y_s} = (1-\beta)\cdot \boldsymbol{y_{hot}} + \frac{\beta}{N}$, where β is a smoothing parameter. With this approach, however, probabilities for all words in the vocabulary are non-zero, including those not present in our target vocabulary. We propose a new vector of probabilities $\mathbf{y}_{sals}$ where for each word we first set the value of non-target words to zero. Among the words in the target vocabulary V_{target}, we compute the semantic similarity of each word $\{v_i\}_{i=1}^{N}$ with other words in the target vocabulary. We use FastText Joulin et al. (2016) to generate word embeddings $\{w_i\}_1^N$ vectors for each word v_i and then compute their cosine similarity. Therefore, we calculate the similarity values as follows:
The final semantically-aware vector of probabilities for word v_i, $\mathbf{y}^i_{sals}$ vector of probabilities will be:

$$\mathbf{y}_{sals} = \frac{1}{Z}\begin{cases} f_{sim}(w_i, w_j) & \geq \lambda \wedge \forall v_j \in V_{target} \\ \frac{\beta}{N} & < \lambda \wedge \forall v_j \in V_{target} \\ 0 & \text{otherwise} \end{cases}$$

Set	Dev						Test					
	BLEU				ROUGE	CHRF++	BLEU				ROUGE	CHRF++
	1	2	3	4			1	2	3	4		
Baselines												
1. Camgoz et al. (2018)	44.40	31.83	24.61	20.16	–	–	44.13	31.47	23.89	19.26	–	–
2. Camgoz et al. (2020)	50.69	38.16	30.53	25.35	–	–	48.90	36.88	29.45	24.54	–	–
3. Yin and Read (2020)	49.05	36.20	28.53	23.52	–	–	47.69	35.52	28.17	23.32	–	–
4. Chen et al. (2022b)	53.57	40.18	31.93	26.40	52.50	49.55	52.81	39.99	31.96	26.43	51.66	49.76
5. Ye et al. (2023)	48.68	37.94	30.58	25.56	–	–	48.30	37.59	30.32	25.54	–	–
Ours												
6. NLLB-Zero Shot	12.26	3.19	1.29	0.64	18.79	19.25	12.71	4.08	1.79	0.84	19.14	19.86
7. NLLB-FineTuned	53.64	40.56	32.35	26.78	53.84	49.53	52.89	40.12	32.03	26.50	53.46	49.65
8. NLLB-Aug	55.12	41.74	33.40	27.76	55.13	50.72	53.63	40.79	32.68	27.13	54.04	50.41
9. NLLB-SALSloss	55.22	42.04	33.56	28.05	55.26	50.65	53.26	40.92	33.00	27.55	54.28	50.01
10. NLLB-all	**55.61**	**42.10**	**33.71**	**28.11**	55.03	**50.64**	**54.79**	**41.90**	**33.77**	**28.20**	**54.44**	**50.79**

Table 1: Comparison with state-of-the-art methods on the PHOENIX-2014T dataset demonstrates the effectiveness of our framework, achieving higher performance despite having approximately a tenth fewer parameters.

Three scenarios can occur: Firstly, for words in the target language with high similarity, defined as λ, we utilize the cosine similarity of their word embeddings, denoted as $f_{sim}(w_i,w_j) = \frac{{w_i}^T \cdot w_j}{\|w_i\|\|w_j\|}$. Secondly, for words with low semantic similarity but present in the target language, we employ standard label smoothing, with β representing the smoothing parameter. Lastly, words outside the target language receive zero smoothing. Subsequently, Z normalizes the vector to sum up to one. One challenge of using this approach with current LLMs lies in the tokenization process, where words may be broken down into subwords by the tokenizer. To address this, we apply semantically aware label smoothing to the initial subword tokens. This method involves comparing the initial token with all other words in the target dataset and increasing the probability of generating similar ones. For subsequent tokens of the same word, target label smoothing is applied, which involves normal smoothing of the labels of target tokens. We use the Semantically Aware Label Smoothing (SALS) in fine-tuning our final model. The $\hat{\mathbf{y}}_{\mathrm{i}}$ represents the output logits corresponding to word v_i, and $\mathbf{y}_{sls}$ denotes the SLS labels. The loss component for a specific class is given by:

$$\ell\left(\hat{\mathbf{y}}_i, \mathbf{y}_{sls}\right) = -\frac{1}{N}\sum_{i=1}^{N} \mathbf{y}_{sls} \log\left(\hat{\mathbf{y}}_i\right)$$

3 Experiments

We evaluate our approach on the PHOENIX-2014T Camgoz et al. (2018) dataset, focusing on German Sign Language videos of weather broadcasts. With 8,257 sequences containing 1,066 glosses and 2,887 German words. This dataset provides a domain-specific benchmark for assessing our fine-tuned models using the BLEU score.

NLLB-200 is a multilingual LLM developed by Meta Costa-jussà et al. (2022), trained on 200 languages. Utilizing *SentencePiece* tokenizer Kudo and Richardson (2018) this model aims to effectively generate and process text in multiple languages, facilitating cross-lingual understanding and generation capabilities. It is trained on a vast corpus comprising 3.6B sentences from low-resource and 40.1B sentences from high-resource languages.

Paraphrasing. For this step, English was chosen as the intermediate translation language using the NLLB-200 model. For each gloss-spoken pair, we translate the spoken language to English and then back to German. We use the 3.3B model with a maximum sequence length of 50 and a beam search of 5 for inference, we generate a total of 7,040 silver label spoken texts and add these gloss-spoken pairs to our primary training dataset.

Back Translation. We generate synthetic glosses by switching the gloss-spoken language pairs to spoken-gloss pairs and fine-tuning a model specifically for this translation task, stopping the training process after 10 epochs. For inference, we pass the training set through the model once more, we add any generated sequences differing from the original gloss to our training set. This augmentation method results in the addition of 6,523 gloss-spoken pairs to our dataset. Consistent with the forward translation, we utilize a maximum sequence length of 100 and a beam search of 5 for inference.

Training. For the Semantically Aware Label Smoothing technique (SALS) we set the cosine similarity threshold to 0.6 to ensure that we consider only words with sufficiently high semantic similarity. We also set β to 0.1. For our final approach, we utilize the NLLB with 3.3B parameters. The architecture consists of 24 encoder and decoder layers. We use the AdamW optimizer Loshchilov and Hutter (2017) to train our network with $\beta_1 = 0.9$ and $\beta_2 = 0.998$. We train the network on two NVIDIA V100 for 60 epochs. Also, our method incorporates a progressive pre-training strategy. Initially, we train the model using a combination of data sources, including augmented data, while applying normal label smoothing. We then fine-tune the model using SALS specifically on the target dataset, ensuring more efficient training and a stronger focus on the task. This progressive approach improves performance by enabling the model to learn from a broad data context before concentrating on the nuances of the target data.

All inferences use a maximum sequence length of 100 and a beam search of 5.

LoRA. To further optimize model performance, we employ the LoRA Hu et al. (2021) technique for fine-tuning. In this approach, we freeze the original model weights and train only the sign language adapter for the target task. This enables us to maintain the original functions of LLMs. Additionally, as demonstrated in Table 2 row 3, LLMs with billions of parameters show the risk of overfitting. By employing the LoRA adapter, we address this concern and can leverage larger models effectively, see row 4. Finally, the final adapter model is memory-efficient, occupying approximately 100 Megabytes of space. For the LoRA configuration, we utilized a rank of 16 and an alpha value of 32.

3.1 Results

Table 1 presents a comparison between our model and previous baselines in terms of BLEU-score. The first baseline Camgoz et al. (2018) employs an RNN encoder-decoder architecture, while the work by Camgoz et al. (2020) utilizes a transformer encoder-decoder trained from scratch. Yin and Read (2020) use a transformer model with FastText embeddings initialization. Chen et al. (2022b) employ a pre-trained multilingual Mbart model, whereas Ye et al. (2023) combine the multilingual Mt5 model and GPT for translation. Our method exhibits superior performance, with 3.75% relative improvement in BLEU-1 (1.98 score diff), 6.69% in BLEU-4 (1.77 score diff), 5.38% in ROUGE (2.78 score diff), and 2.07% in CHRF++ (1.03 score diff), with significantly fewer trainable parameters(appendix A), enhancing both effectiveness and efficiency. Additionally, we observe a performance discrepancy between the dev and test sets in some previous models, as seen in the first three rows. However, our method demonstrates an average score decrease of only 0.22 across BLEU, ROUGE, and CHRF++ metrics when transitioning from the development set to the test set. Our experiments indicate that our label smoothing method contributes to improved generalization, effectively minimizing this performance gap.

	BLEU-1	BLEU-4
600M	52.7	26.5
1.3B	53.4	27.3
3.3B	53.1	27.1
3.3B+LoRA	53.8	27.5

Table 2: Comparison of BLEU-score performance across different model sizes, ranging from 600M to 3.3B parameters, for gloss translation tasks.

Zero Shot performance: To further evaluate the understanding of these models for the sign language translation task, we conduct an experiment without fine-tuning the Language Models (LLMs), creating a zero-shot scenario. As shown in Table 1 row 6, despite the prior training of this model on the German data, the results proved to be suboptimal. This underscores the main role of fine-tuning in optimizing LLMs for *Gloss2Text* translation.

Loss Function: To evaluate the effectiveness of our modified loss function, we conduct two set experiments. Initially, we exclude our SALS term during fine-tuning, substituting it with conventional cross-entropy loss, row 7. Subsequently, we replace the cross-entropy loss with our proposed loss for comparison, row 9. Table 1, shows that our model demonstrates better performance with the integration of semantically aware label smoothing (see rows 7 and 9). The NLLB-SALSloss system demonstrates an average improvement of 1.68 points on the development set and 1.06 points on the test set over the NLLB-FineTuned system across BLEU, ROUGE, and CHRF++ metrics.

Data Augmentation Techniques: We also compare our method with various data augmentations, namely paraphrasing and backward translation, for the gloss translation task. Table 1, row 8, demonstrates the improvements achieved by applying these augmentations using the cross-entropy loss.

Model Size: We utilize NLLB-200 models ranging from 600M to 3.3B parameters. As illustrated in Table 2, larger models generally lead to bet-

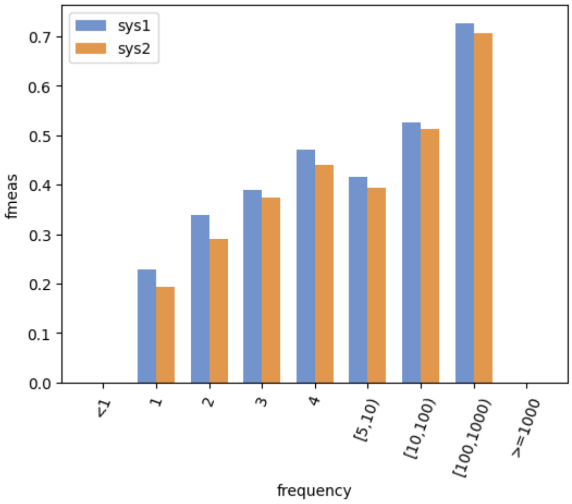


Figure 3: Comparison of word-level F-measure scores across different frequency buckets with Chen et al. (2022b).

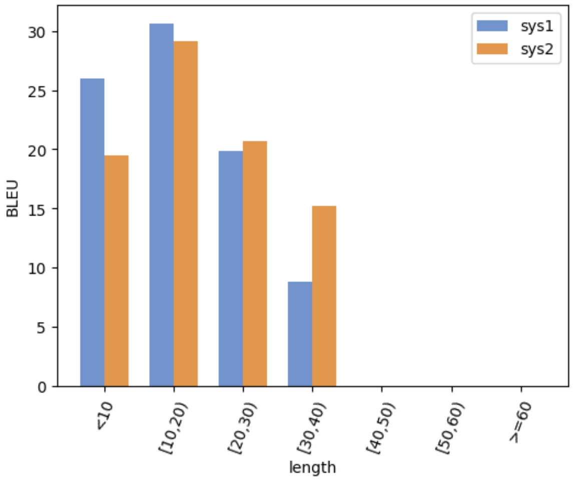


Figure 4: BLEU score vs. sentence length, showing our approach's translation (sys 1) quality decreases with longer sentences.

ter translation performance. However, with the largest model, we observed signs of overfitting to the dataset. The reason is the fine-tuning dataset is relatively small compared to the number of parameters in the model. To address this issue, we explore LoRA techniques to enhance model performance and mitigate overfitting.

4 Analysis with previous SOTA

We compare our results ("sys 1") with the previous state-of-the-art method Chen et al. (2022b). Our initial comparison focuses on the length of the predictions. It appears that our method generates shorter sentences compared to the more lengthy sentences produced by the previous state-of-the-art. Notably, both methods used the same hyperparameters for generation: a length penalty of 1, a maximum length of 100, and a beam search size of 5. Our method's generation ratio is $0.9808(ref = 8458, out = 8296)$, whereas the ratio for Chen et al. (2022b) is $1.0233(ref = 8458, out = 8655)$.

Word Accuracies: We also evaluated word-level F-measure scores across different frequency buckets. This analysis allows us to observe the performance of our method relative to the frequency of the words in the dataset. Our method shows consistent improvements across most frequency buckets compared to the previous state-of-the-art. This suggests that our approach is more effective at predicting both common and rare words accurately. Detailed results are shown in Figure 3. We further analyzed the performance of our method by comparing sentence-level F-measure scores across different sentence-length buckets. The results indicate that our method excels in predicting shorter sentences but lags behind in longer sentences, likely due to the other method generating lengthier predictions. The Figure 4 illustrates These findings.

Translation examples: Tables 6 and 7 in the supplementary material present several translation examples along with their BLEU scores. The first table highlights instances where our translations outperform, while the second table showcases failure cases where the previous state-of-the-art methods perform better.

5 Conclusion

We conducted a comprehensive exploration of *Gloss2Txt* translation for sign language using large language models and the PHOENIX-2014T dataset. We evaluated different model architectures, data augmentations, and loss functions. Our experiments showed that our Semantically Aware Label Smoothing technique significantly improves translation quality over state-of-the-art models.

Limitations and Open problems

While glosses provide additional structured annotation of sign language videos, they do not fully capture the complexity of sign language communication. Facial expressions, which are part of conveying meaning in sign language, are often not represented in gloss annotations. Additionally, gestures involving pointing to specific locations or objects are typically omitted in gloss representations. Even with the visual modality, such expressive nuances are often absent in text gloss representations, but are captured in the corresponding spoken language translations.

Another limitation is the computational overhead introduced by our approach. The method requires calculating similarities between the gloss and target vocabularies, which can slow down the process. This issue becomes more pronounced as the vocabulary size increases, making the training stage both slower and more resource-intensive.

Also, it's important to acknowledge that datasets used in sign language often have a domain-specific vocabulary. This vocabulary may not always reflect the everyday activities and interactions prevalent in the deaf community. This potentially limits the scope and applicability of sign language systems developed using such datasets and calls for additional benchmarks and evaluation methodologies for sign language translation.

Acknowledgements

This work was partially supported by the Amazon Research Award and resources provided by the Office of Research Computing at George Mason University (URL: https://orc.gmu.edu) and funded in part by grants from the National Science Foundation (Award Number 2018631). Antonios Anastasopoulos is additionally generously supported by the National Science Foundation under grant IIS-2327143.

References

Galina Angelova, Eleftherios Avramidis, and Sebastian Möller. 2022. Using neural machine translation methods for sign language translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 273–284.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10023–10033.

Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen Lin. 2022a. A simple multi-modality transfer learning baseline for sign language translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5120–5130.

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. 2022b. Two-stream network for sign language recognition and translation. *Advances in Neural Information Processing Systems*, 35:17043–17056.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pretraining for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. 2019. When does label smoothing help? *Advances in neural information processing systems*, 32.

Graham Neubig, Zi-Yi Dou, Junjie Hu, Paul Michel, Danish Pruthi, Xinyi Wang, and John Wieting. 2019. compare-mt: A tool for holistic comparison of language generation systems. *CoRR*, abs/1903.07926.

Rico Sennrich and Biao Zhang. 2019. Revisiting low-resource neural machine translation: A case study. *arXiv preprint arXiv:1905.11901*.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Jinhui Ye, Wenxiang Jiao, Xing Wang, and Zhaopeng Tu. 2023. Scaling back-translation with domain text generation for sign language gloss translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 463–476.

Kayo Yin and Jesse Read. 2020. Better sign language translation with STMC-transformer. *arXiv preprint arXiv:2004.00588*.

Xuan Zhang and Kevin Duh. 2021. Approaching sign language gloss translation as a low-resource machine translation task. In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, pages 60–70.

Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20871–20881.

Dele Zhu, Vera Czehmann, and Eleftherios Avramidis. 2023. Neural machine translation methods for translating text to sign language glosses. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12523–12541.

A Trainable parameters

In comparison with previous SOTA methods, we also investigate the total number of trainable parameters. Although our method uses a larger model, it does not require many parameters to train due to the use of LoRA adapters. This allows the original LLM to be used for other tasks, with the adapter applied only when sign language gloss translation is needed. In our experiments, all the linear layers are used in the LoRA fine-tuning for optimal performance. Table 3 compares the trainable parameters.

	Trainable Params
Camgoz et al. (2018)	100M
Camgoz et al. (2020)	150M
Yin and Read (2020)	130M
Chen et al. (2022b)	611M
Ye et al. (2023)	570M
NLLB-200	26M

Table 3: Comparison of model sizes and the number of trainable parameters for various methods, highlighting the efficiency of our approach with LoRA adapters.

	#Params	BLEU-1	BLEU-4
GPT-2 Scratch	124M	45.21	19.3
GPT-2(German)	137M	49.4	24.94
M-Bart-25	610M	52.1	26.4
MT5-Base	580M	50.1	24.4
NLLB-200	600M	52.7	26.5

Table 4: Comparison of performance across different architecture types, including encoder-decoder and decoder-only models, for gloss translation tasks. For a fair comparison, pre-trained models were selected to have approximately the same number of parameters across all architecture types.

B Architecture Type

We initially investigated various architectures for the task of gloss translation. To our knowledge, this is the first attempt to compare the impact of Large Language Model LLM architectures on gloss translation. We explore both decoder-only models, specifically GPT-2, and encoder-decoder models, including MT5-Base, Mbart-50, and NLLB-200. The results presented in Table 4 illustrate their performance. We use the NLLB with 600M parameters to closely match the others in terms of trainable parameters. Our experiments show that encoder-decoder models perform better than the GPT model. This could potentially be attributed to the diverse pretraining datasets utilized by encoder-decoder models, enabling them to comprehend low-resource tasks more effectively and leverage their multilingual capabilities.

C Back Translation model

The quality of the back-translated model directly correlates with the quality of the generated data and, consequently, influences the final performance. Table 5 presents a comparison between two models, both trained with the same configuration. As depicted, the NLLB-200 model also surpasses the Mbart-25 in the back translation task. Through fine-tuning, the NLLB-200 model generates higher-quality pseudo-parallel data, as shown by better BLEU scores on the validation set of the text-to-gloss translation.

	BLEU-1	BLEU-4
Nbart-50	64.57	25.48
NLLB-200 Aug	67.63	26.47

Table 5: Back translation performance under various fine-tuning approaches. NLLB-200 archives better performance in generating pseudo-gloss annotations for the text-to-gloss Dev set.

Reference	später ist es meist trocken.	BLEU
Chen et al. (2022b)	später wird es aber schon wieder trockener.	18.27
Ours	später ist es meist trocken.	100
Reference	dort morgen bis zweiundzwanzig grad.	
Chen et al. (2022b)	morgen temperaturen von zweiundzwanzig grad im breisgau bis zweiundzwanzig grad am oberrhein.	19.14
Ours	dort morgen bis zweiundzwanzig grad.	100
Reference	der deutsche wetterdienst hat entsprechende warnungen herausgegeben.	
Chen et al. (2022b)	es gelten entsprechende warnungen des deutschen wetterdienstes.	21.73
Ours	der deutsche wetterdienst hat entsprechende warnungen herausgegeben.	100
Reference	jetzt wünsche ich ihnen noch einen schönen abend.	
Chen et al. (2022b)	guten abend liebe zuschauer.	12.83
Ours	und jetzt wünsche ich ihnen noch einen schönen abend.	89.09
Reference	auf den bergen sind orkanartige böen möglich.	
Chen et al. (2022b)	im bergland sind zum teil orkanartige böen möglich.	40.32
Ours	auf den bergen sind orkanartige böen möglich.	100
Reference	am montag meist trocken bei einer mischung aus sonne und wolken.	
Chen et al. (2022b)	am montag ist es meist trocken sonne und wolken gibt es eine mischung aus nebel und sonne.	19.20
Ours	am montag bleibt es meist trocken bei einer mischung aus sonne und wolken.	74.66
Reference	der deutsche wetterdienst hat entsprechende unwetterwarnungen herausgegeben.	
Chen et al. (2022b)	es gelten entsprechende warnungen des deutschen wetterdienstes.	16.51
Ours	der deutsche wetterdienst hat entsprechende warnungen herausgegeben.	65.80
Reference	ähnliches wetter auch am donnerstag.	
Chen et al. (2022b)	und nun die wettervorhersage für morgen donnerstag den achten juli.	11.64
Ours	ähnliches wetter dann auch am donnerstag.	59.15
Reference	jetzt wünsche ich ihnen noch einen schönen abend.	
Chen et al. (2022b)	ihnen noch einen schönen abend und machen sie es gut.	42.64
Ours	und jetzt wünsche ich ihnen noch einen schönen abend.	89.09
Reference	heute nacht liegen die werte zwischen vierzehn und sieben grad.	
Chen et al. (2022b)	heute nacht vierzehn bis sieben grad.	24.00
Ours	heute nacht werte zwischen vierzehn und sieben grad.	66.51

Table 6: Here are example translations comparing our method with the previous state of the art Chen et al. (2022b). These examples, provided by Neubig et al. (2019), highlight instances where our method achieves a higher BLEU score.

Reference	auch am tag wieder viel sonnenschein später bilden sich hier und da ein paar quellwolken.	BLEU
Chen et al. (2022b)	auch am tag viel sonne später hier und da ein paar quellwolken.	50.93
Ours	auch am tag scheint verbreitet die sonne später kommen an den küsten wieder dichtere wolken auf.	15.09
Reference	sonst ist es recht freundlich.	
Chen et al. (2022b)	sonst ist es recht freundlich.	100
Ours	ansonsten wird es recht freundlich.	60.42
Reference	im südosten regnet es teilweise länger.	
Chen et al. (2022b)	im südosten regnet es teilweise ergiebig.	70.34
Ours	in der südosthälfte regnet es teilweise ergiebig.	30.73
Reference	im süden bleibt es morgen unter hochdruckeinfluss zunächst noch recht freundlich und warm.	
Chen et al. (2022b)	im süden deutschlands bleibt es morgen unter hochdruckeinfluss noch weitgehend freundlich und warm.	53.95
Ours	in der südhälfte bestimmt hochdruckeinfluss morgen unser wetter und es bleibt noch ziemlich warm.	14.05
Reference	morgen reichen die temperaturen von einem grad im vogtland bis neun grad am oberrhein.	
Chen et al. (2022b)	morgen reichen die temperaturen von einem grad im vogtland bis neun grad am oberrhein.	100
Ours	morgen temperaturen im vogtland bis neun grad.	27.09
Reference	auch am tag wieder viel sonnenschein später bilden sich hier und da ein paar quellwolken.	
Chen et al. (2022b)	auch am tag viel sonne später hier und da ein paar quellwolken.	50.93
Ours	auch am tag scheint verbreitet die sonne später kommen an den küsten wieder dichtere wolken auf.	15.09
Reference	morgen vormittag an der ostsee noch starke böen sonst weht der wind schwach bis mäßig aus ost bis südost.	
Chen et al. (2022b)	morgen vormittag an der nordsee noch kräftige böen sonst weht der wind schwach bis mäßig.	50.76
Ours	morgen vormittags an der nordsee starke bis stürmische böen sonst meist nur schwacher bis mäßiger wind aus süd bis südwest.	11.84
Reference	morgen muss verbreitet mit teilweise kräftigen schauern und gewittern gerechnet werden.	
Chen et al. (2022b)	morgen muss mit teilweise unwetterartigen schauern und gewittern gerechnet werden.	56.53
Ours	morgen gibt es dort zum teil kräftige schauer und gewitter.	11.76
Reference	sonst viel sonnenschein.	
Chen et al. (2022b)	sonst viel sonnenschein.	100
Ours	ansonsten scheint verbreitet die sonne.	19.30

Table 7: Here are example translations of failed cases where our method obtains a lower BLEU score.