POSESTITCH-SLT: Linguistically Inspired Pose-Stitching for End-to-End Sign Language Translation

Abhinav Joshi* **Vaibhav Sharma*** **Sanjeet Singh*** **Ashutosh Modi**
Department of Computer Science and Engineering
Indian Institute of Technology Kanpur (IIT Kanpur)
{ajoshi, svaibhav, sanjeet, ashutoshm}@cse.iitk.ac.in

Abstract

Sign language translation remains a challenging task due to the scarcity of large-scale, sentence-aligned datasets. Prior arts have focused on various feature extraction and architectural changes to support neural machine translation for sign languages. We propose **POSESTITCH-SLT**, a novel pre-training scheme that is inspired by linguistic-templates-based sentence generation technique. With translation comparison on two sign language datasets, How2Sign and iSign, we show that a simple transformer-based encoder-decoder architecture outperforms the prior art when considering template-generated sentence pairs in training. We achieve BLEU-4 score improvements from 1.97 to 4.56 on How2Sign and from 0.55 to 3.43 on iSign, surpassing prior state-of-the-art methods for pose-based gloss-free translation. The results demonstrate the effectiveness of template-driven synthetic supervision in low-resource sign language settings.

1 Introduction

Sign languages are the primary mode of communication for over 70 million people from the deaf and hard of hearing (DHH) community globally, according to the World Federation of the Deaf (Tran et al., 2024). Despite increasing progress in natural language processing (NLP) (Wang et al., 2019), sign language processing remains significantly underexplored, both in terms of benchmark datasets and model development (Min et al.; Yin et al., 2021; Moryossef et al., 2020; Jiang et al., 2024). In contrast to spoken languages, sign languages are visual-gestural and multimodal, combining manual signs (e.g., hand shapes and movements) with non-manual cues (e.g., facial expressions and body posture) (Cohn, 2013). Moreover, training on raw sign videos raises fairness and privacy concerns due to signer-identifiable features. Pose-based approaches (Ko et al., 2019; Camgoz et al., 2020), which use 2D/3D keypoints from the face, hands, and body, provide a privacy-preserving alternative while retaining essential communication information.

In this work, we focus on pose-based, gloss-free Sign Language Translation (SLT) under real-world constraints of data scarcity and signer privacy. To address the lack of large-scale parallel data, we propose **POSESTITCH-SLT**: a **linguistically inspired pretraining** strategy that constructs synthetic pose-based sentence data, making use of publicly available word-level sign language datasets and linguistic templates (Warstadt et al., 2020) to generate millions of grammatically diverse sentences in English. We use vocabularies from CISLR (Joshi et al., 2022) for Indian Sign Language (ISL) and WLASL for American Sign Language (ASL) (Li et al.), two of the largest publicly available word-level sign datasets, covering approximately 4.5K and 2K words, respectively. While these vocabularies remain limited compared to spoken-language corpora, they represent the most extensive resources currently accessible to the SLT research community. This choice was made intentionally to enable scalable pretraining without relying on proprietary datasets or gloss annotations. Furthermore, our framework is modular and adaptable, designed to make the most of available resources while remaining extensible as new sign language datasets become available. Fig.1 illustrates the entire pipeline. Importantly, we employ a standard transformer architecture (Vaswani et al., 2017) by design, in order to isolate and rigorously assess the impact of the proposed pretraining strategy. This design choice ensures that performance improvements are not confounded by model-specific enhancements, and highlights the generality of our approach. We evaluate our method on two challenging, publicly available benchmarks suitable for gloss-free, pose-only translation: How2Sign (ASL) (Duarte et al., 2021) and iSign (ISL) (Joshi et al.,

* Equal Contributions

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 13846–13865
November 4-9, 2025

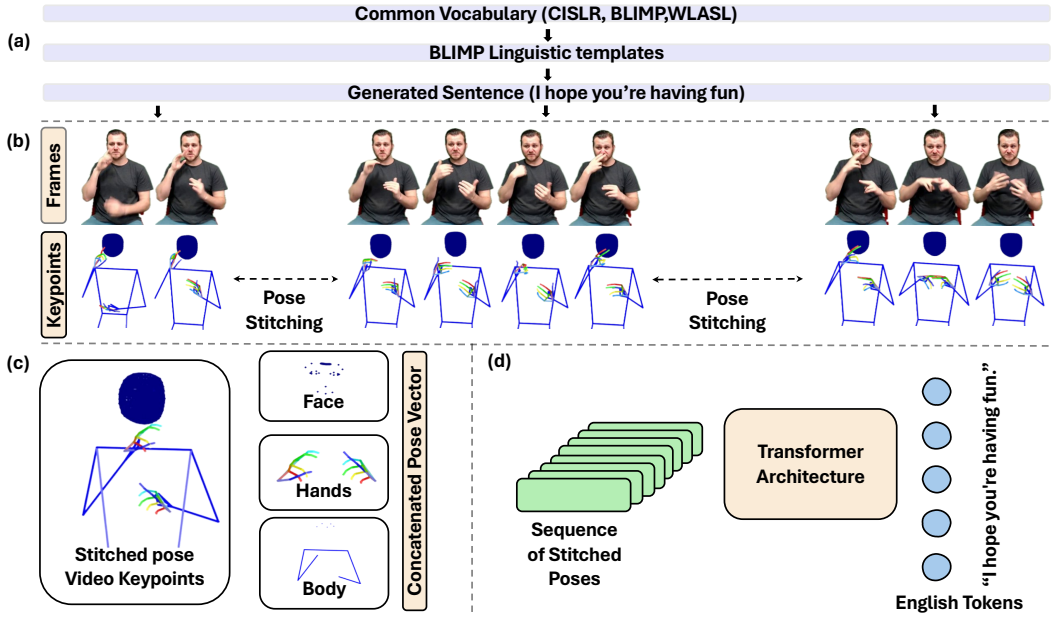

Figure 1: The figure shows the **POSESTITCH-SLT** pipeline for generating a Sign Language Dataset for translation using pose stitching based on linguistic templates. Starting with a common vocabulary shared across datasets like CISLR, BLIMP, and WLASL, sentences are generated using BLIMP linguistic templates (Warstadt et al., 2020). For each generated sentence, corresponding sign language poses for words/gloss are stitched together. Frames are extracted from the stitched pose video to form a sequence of keypoints. These keypoints extracted from the face, hands, and body are concatenated to create pose vectors. This sequence of pose vectors serves as input to an encoder-decoder-based transformer model. The sentence example is taken from the How2Sign dataset, saying, "I hope you're having fun".

2024). In a nutshell, we make the following contributions:

- We introduce **POSESTITCH-SLT**, a ***linguistically grounded strategy*** for synthetic dataset creation and model pretraining in ***pose-based, gloss-free sign language translation.***
- We demonstrate ***state-of-the-art results***, with BLEU-4 gains of ***1.97–4.56 on How2Sign*** and ***0.55–3.43 on iSign***, using only pose inputs in a gloss-free setting and a standard Transformer architecture.
- We release the dataset and code via the GitHub: `https://github.com/Exploration-Lab/PoseStich-SLT`.

2 Related Work

SLT has gained increasing interest in recent years (Camgoz et al., 2018; De Coster and Dambre, 2022; De Coster et al., 2021; Chen et al., 2022b; Joshi et al., 2023). Most prior works (Yin and Read, 2020; Moryossef et al., 2021) rely heavily on intermediate gloss annotations to enable supervised training of translation systems. The gloss-based approaches assume a two-stage pipeline: first mapping sign videos to glosses, and then translating glosses into spoken language. While effective, they require extensive manual annotation and may strip away linguistic richness. Early work in end-to-end sign language translation, such as STMC-Transformer (Yin and Read, 2020), extends the STMC architecture originally developed for gloss-level recognition (Zhou et al., 2022). However, a central challenge across these models is the token length mismatch between input sign video frames and output textual tokens (Lin et al., 2023; Cihan Camgoz et al., 2020; Álvarez et al.). This issue complicates sequence alignment, especially in encoder-decoder frameworks, where long visual input sequences must be compressed to shorter textual outputs. To mitigate this mismatch, several approaches have been proposed. Some rely on learning intermediate gloss-like representations through frame clustering or joint training objectives (Cihan Camgoz et al., 2020; Chen et al., 2022a), while others (Ahn et al.; Yin and Read, 2020) modify the architecture by introducing conceptual anchors or applying CTC loss for better temporal alignment (Graves et al., 2006). GloFE (Lin et al., 2023) is a recent gloss-free system that addresses these alignment challenges using pose-based inputs. It introduces weak intermediate representations derived from spoken-language tokens (termed "conceptual anchors") to supervise visual feature learn-

ing. GloFE is closely related to our target setting, as it also uses keypoint-based inputs and bypasses gloss labels. Our work differs in both approach and emphasis. Instead of relying on architectural modifications or auxiliary representations, we propose a linguistically driven pretraining strategy that leverages publicly available word-level sign language datasets. Using grammar templates from BLiMP, we synthesize millions of sentence-level training examples by stitching pose sequences from CISLR (ISL) (Joshi et al., 2022) and WLASL (ASL) (Li et al.). Despite using a vanilla encoder-decoder Transformer, our model achieves performance comparable to or better than specialized models like GloFE, demonstrating the effectiveness and scalability of our approach.

3 Methodology

Problem Setup: We consider the task of translating sign language videos, represented as 2D pose sequences, into spoken language text. Formally, given a sequence of frame-wise pose vectors $\mathcal{X} = \{x_1, x_2, \ldots, x_T\}$ extracted from a sign language video, where each $x_t \in \mathbb{R}^{152}$ is a concatenation of keypoints from the face, hands, and upper body, the goal is to generate a corresponding English sentence $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\}$. In contrast to prior work that depends on intermediate gloss annotations or signer-identifiable video, we directly learn from pose-level inputs to preserve privacy and avoid costly manual labeling.

Data Generation via Linguistic Templates and Large Text Corpora: To overcome the dataset scarcity, we construct synthetic training datasets by leveraging grammatical templates covering a wide range of linguistic phenomena. Our key insight is that by aligning linguistic templates with word-level sign pose data, we can synthesize large numbers of sentence-pose pairs suitable for model pretraining. We take inspiration from the BLiMP benchmark (Warstadt et al., 2020), which captures a broad set of linguistic phenomena (67 paradigms across 12 categories). Using these templates, we generate millions of grammatically diverse English sentences. However, to synthesize paired sign sequences, we require corresponding pose representations for each word in the targeted sign language. To this end, we leverage two publicly available word-level sign language datasets: WLASL (Li et al.) for American Sign Language (ASL) and CISLR (Joshi et al., 2022) for Indian Sign Language (ISL). We identify the overlapping vocabulary between BLiMP and each of these datasets, 508 words for WLASL and 504 for CISLR, and construct two synthetic datasets: **1) BLiMP-ASL: *2.8 million*** English sentences using the shared vocabulary between WLASL and BLiMP. **2) BLiMP-ISL: *22 million*** English sentences using the shared vocabulary between CISLR and BLiMP. For example, consider the Adjunct Island paradigm from BLiMP Templates (Warstadt et al., 2020):

`Wh[] Aux_mat[] Subj[] V_mat[] Adv[] V_emb[] Obj[]`

By filling this template with shared vocabulary, we can obtain sentences like: "`What did John read before filing the book?`" In this example, the words like "`what,`" "`John,`" "`read,`" "`filing,`" and "`book,`" are drawn from the overlapping vocabulary of BLiMP–CISLR or BLiMP–WLASL and the syntactic structure is governed by the BLiMP template. Our sentence generation process combines linguistically motivated templates with sign language vocabulary. Since sign languages are less studied in the literature and we do not yet have a comprehensive understanding of sign language grammar, particularly in low-resource settings such as ISL, we ground the sentence generation process in English grammar, under the assumption that there exist underlying correlations between spoken and sign language grammars. In doing so, we ensure that the resulting sentences are both grammatically well-formed and systematically associated with specific linguistic phenomena. (see App. A.1 for more details).

Though the BLiMP-based generation provides strong grammatical diversity, its vocabulary coverage is limited. To increase linguistic and lexical coverage, we complement this with data from the BPCC corpus (Gala et al., 2023), a large collection of ***230 million*** English bitext pairs. From this, we select sentences that have a word match of over 90% with the WLASL or CISLR vocabulary, thereby ensuring that they can be fully synthesized using the available sign poses. This yields two additional datasets: BPCC-ASL and BPCC-ISL. Further post-processing (e.g., sentence length matching (see App. A.3 for more details)., anonymization via token replacement (see App. A.4 for more details), and filtering infrequent words) ensures compatibility with downstream training objectives. Further details and dataset statistics are presented in App. A.

Pose Stitching: To synthesize sentence-level sign language sequences, we stitch together word-level

	Method	Source	DEV				TEST			
			BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEU-1	BLEU-2	BLEU-3	BLEU-4
How2Sign (ASL)	Alvarez	–	17.73	7.94	4.13	2.24	17.40	7.69	3.97	2.21
	GloFE	–	14.92	7.13	3.90	2.27	14.86	6.99	3.64	1.97
	w/o Pose Stitched	–	22.19	9.71	5.43	3.37	18.16	7.96	4.33	2.62
	Pose Stitched (RWO)	BPCC	26.25	13.00	7.50	4.62	26.70	**12.90**	7.23	4.32
		BLIMP	26.22	13.06	7.81	**5.04**	25.17	12.26	7.15	4.53
	Pose Stitched (SWO)	BPCC	**27.35**	**13.56**	7.85	4.95	**26.74**	12.86	**7.26**	4.44
		BLIMP	26.89	13.36	**7.92**	**5.04**	25.98	12.55	7.25	**4.56**
	Best (Ours)	–	**27.35 (+9.62)**	**13.56 (+5.62)**	**7.92 (+3.79)**	**5.04 (+2.77)**	**26.74 (+9.34)**	**12.90 (+5.21)**	**7.26 (+3.29)**	**4.56 (+2.35)**
iSign (ISL)	GloFE	–	8.92	2.85	1.18	0.61	9.20	2.83	1.12	0.55
	w/o Pose Stitched	–	12.85	2.50	1.04	0.58	12.81	2.66	1.14	0.64
	Pose Stitched (RWO)	BPCC	14.75	6.11	3.60	2.45	14.89	6.25	3.67	2.51
		BLIMP	13.89	4.86	2.57	1.62	13.39	4.63	2.43	1.48
	Pose Stitched (SWO)	BPCC	**17.31**	**8.09**	**5.02**	**3.54**	**17.67**	**8.20**	**5.00**	**3.43**
		BLIMP	16.42	7.51	4.62	3.23	16.44	7.40	4.49	3.09
	Best (Ours)	–	**17.31 (+8.39)**	**8.09 (+5.24)**	**5.02 (+3.84)**	**3.54 (+2.93)**	**17.67 (+8.47)**	**8.20 (+5.37)**	**5.00 (+3.88)**	**3.43 (+2.88)**

Table 1: BLEU score results on the How2Sign and iSign datasets comparing different pretraining strategies (random word order vs. same word order), along with baseline results (GloFE and Alvarez). Corresponding ROUGE scores are provided in App. Table 12. The numbers in brackets show the absolute improvements from the baseline.

pose sequences. For each sentence, we retrieve individual word videos from WLASL or CISLR and extract 2D keypoints using the Mediapipe library (MediaPipe, 2023), covering facial expressions, hand configurations, and upper body motion. We select 76 keypoints (forming 152-dimensional vectors) that most effectively capture sign-relevant articulation, inspired by prior work (Lin et al., 2023). Low-confidence keypoints are interpolated to maintain consistency, and all sequences are normalized to reduce inter-signer variance.

The stitched pose sequences are constructed by temporally aligning and concatenating word-level segments into a fluent stream. For smooth transitions between signs and to avoid abrupt motion boundaries, we apply boundary-aware temporal smoothing (Bulas-Cruz et al.) The resulting pose sequences emulate coherent signing while retaining compositional structure.

A key design choice is the word order used during pose stitching. Unlike spoken languages, most sign languages have a distinct and often non-linear grammatical structure that differs significantly from English. However, due to the lack of accessible linguistic resources, reliable syntactic parsers, or annotated corpora for ASL or ISL grammar, we do not attempt to reconstruct native sign language word order in our synthetic data. Instead, we adopt English word order as a proxy, which simplifies generation and leverages the available textual infrastructure. Further details regarding framerate matching and Pose processing are discussed in the App. B.1 , B.2 .

To explore how sensitive the model is to this assumption, we construct two variants of our synthetic datasets: **1)** ***Same Word Order (SWO):*** Poses are stitched in the same order as the English sentence, preserving syntactic structure and compositional cues. **2)** ***Random Word Order (RWO):*** Poses are stitched after randomly permuting the word order, injecting syntactic noise, and encouraging the model to learn flexible and robust representations. These two variants allow us to investigate the trade-off between syntactic alignment and generalization, especially in low-resource or cross-lingual sign language translation settings.

To effectively leverage the synthetic datasets and ensure a smooth transition to real-world data, we also employ a linear annealing strategy during training (see App. C.2 for more details). Initially, the model is trained exclusively on synthetic pose-sentence pairs. As training progresses, we gradually increase the probability of sampling from the real sentence-aligned datasets, iSign for ISL and How2Sign for ASL, up to a threshold of 85% at 60,000 training steps (also see App. 20). After this point, training continues predominantly on real data (sign language pose-sequences from iSign and How2Sign datasets for ISL and ASL, respectively). This staged training approach helps the model benefit from both the diversity and scale of synthetic data and the realism of target domain data. We found that this strategy works better than the traditional disjoint pretraining and fine-tuning phases. Unlike traditional pretraining-fine-tuning pipelines with disjoint phases, our curriculum is blended and progressive, allowing for continual adaptation and preventing catastrophic forgetting. The full architecture (Transformer-based encoder-decoder), tokenization strategy, and hyperparameters are de-

tailed in the App. C.

4 Results and Analysis

To evaluate the effectiveness of our synthetic pose-based pretraining strategy, we train a Transformer encoder-decoder model (Vaswani et al., 2017) on two benchmark datasets: How2Sign (ASL) and iSign (ISL).

Translation Performance: We report BLEU-4 (Papineni et al., 2002) and ROUGE-L (Lin, 2004) scores for both datasets, following evaluation protocols from prior work (Lin et al., 2023) to ensure fair comparison. As shown in Table 1, our approach outperforms the previous state-of-the-art, GloFE, by a substantial margin: On How2Sign, BLEU-4 improves from $2.27 \rightarrow 5.04$ on the dev set and $1.97 \rightarrow 4.56$ on the test set. On iSign, BLEU-4 improves from $0.61 \rightarrow 3.54$ on the dev set and $0.55 \rightarrow 3.43$ on the test set. We also observe consistent improvements in ROUGE scores across both datasets (see App. Table 12), further validating the quality of generated translations. We also report the SacreBLEU scores of the best model (see App. Table 13). While the scores remain modest, the consistent gains demonstrate the promise of our approach for improving SLT in low-resource sign languages.

Ablation/Impact of Synthetic Data: To isolate the impact of synthetic pose-stitched pretraining, we conduct a baseline experiment using the same architecture and hyperparameters, but ***without any synthetic data***. As shown in App. Fig. 18, this variant performs significantly worse, highlighting the critical role of the proposed pretraining pipeline.

Generalization to Unseen Sentences: To test generalization, we evaluate our pretrained models on new pose-stitched sentences not seen during training. Surprisingly, the model achieves BLEU-4 scores of 97 and 47 on two synthetic evaluation sets (App. Fig. 19), indicating that the model learns to robustly handle sentence generation within the restricted vocabulary domain.

Similarity to Target Domain: To further understand domain alignment, we compute semantic similarity scores between sentences in the synthetic and target datasets using SBERT (all-MiniLM-L6-v2) (hug). App. Table 7 shows that higher-similarity examples correlate with better translation quality, reinforcing the effectiveness of our dataset construction approach.

Qualitative Results: App. Table 10 presents qualitative examples of predicted translations alongside ground truth references. Our model consistently captures the main semantic content and shows better alignment than GloFE. However, occasional issues such as phrase repetition or hallucinations remain, likely due to synthetic data noise. Additional results are provided in the App. D . The effect of adding the pose-stitched dataset is discussed in App. D.1, while the effect of distribution shift is analyzed in App. D.2. Qualitative analyses and the impact of random versus same word order in the pretraining dataset are presented in the App. D.3 and App. D.4.

Grammatical Notion of Generated Sign Language Sentences: Understanding and modeling sign language grammar is a difficult task. Sign language relies on visual-spatial structures, non-manual markers, role shifts, and the use of signing space, making its grammar highly complex (Sinha, 2017; Joshi et al., 2024). Standardized grammatical resources for low-resource languages like ISL are limited. Similar challenges are noted for ASL, where resources such as (Sehyr et al., 2021) and recent work (Tavella et al., 2022) only begin to capture aspects of its grammar. These complexities highlight that directly modeling sign language grammar is non-trivial. This motivates us to ground synthetic data generation in English grammatical templates while acknowledging the gap between spoken and signed language structures.

5 Conclusion and Future Directions

In this work, we introduce a novel training paradigm for sign language translation by leveraging existing word-level pose datasets to synthesize sentence-level training data. Our approach constructs pose-stitched sentence sequences using linguistically grounded templates, enabling large-scale pretraining without requiring expensive gloss annotations or raw video footage. Through extensive experiments on two benchmark datasets, How2Sign (ASL) and iSign (ISL), we demonstrate that this strategy significantly improves translation performance, even when using a standard Transformer-based encoder-decoder model.

This work opens new avenues for scaling sign language translation using linguistic structure-based data synthesis. Future efforts may explore expanding the vocabulary coverage of word-level pose datasets, integrating grammatical features from sign languages, and applying this strategy across more diverse sign language variants to build more inclusive and generalizable SLT systems.

Limitations

Despite the improvements demonstrated by our training strategy, showing a significant boost in performance when compared to existing SOTA methods, there are several limitations that remain open:

Vocabulary Coverage: Our method relies on the intersection of BLiMP vocabulary and available word-level sign datasets (WLASL and CISLR), which cover only 2K–4.5K words. This restricts the expressiveness of synthetic sentences and limits its generalization. Extending word-level datasets to include broader and more diverse vocabularies remains critical, but was beyond the scope of this work due to a lack of publicly available resources.

Absence of Sign Language Grammar: We rely on English word order for generating pose-stitched sequences, as grammatical annotations for sign languages are limited and not standardized. Although this introduces potential mismatches, modeling sign-specific syntax would require extensive linguistic resources and annotation efforts, which are currently lacking for most sign languages.

Architectural Simplicity: We adopt a standard transformer model to isolate the effect of our training strategy. While this ensures clarity in evaluation, it may underutilize recent advances in sign-specific architectures. This choice was made intentionally to enable scalable pretraining without relying on proprietary datasets or gloss annotations. Moreover, we could not fully replicate prior methods (e.g., GloFE) due to unavailable details, limiting direct comparisons. Furthermore, our framework is modular and adaptable, designed to make the most of available resources while remaining extensible as new sign language datasets become available, which further adds to the advantage of using simple architectures.

Generality Across Languages: Although our approach is applicable beyond ASL and ISL, broader adoption depends on the availability of word-level datasets in other sign languages. Resource creation in this space remains a foundational challenge.

Ethical Considerations

Our work focuses on the task of sign language translation, with an emphasis on both American Sign Language (ASL) and Indian Sign Language (ISL). The goal is to use technology to enhance the daily lives of the deaf and hard-of-hearing communities in both regions. While we have made improvements over previous models, the proposed system still lacks the capability to function as a fully realized interpreter in real-life scenarios. We use extracted key points as the input for the model, ensuring minimal to no concerns regarding personal privacy.

Acknowledgments

We would like to thank the anonymous reviewers and the meta-reviewer for their insightful comments and suggestions. This research work was partially supported by the Research-I Foundation of the Department of CSE at IIT Kanpur.

References

Sentence-transformers/all-MiniLM-L6-v2 · Hugging Face — huggingface.co. https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2. [Accessed 09-09-2025].

Junseok Ahn, Youngjoon Jang, and Joon Son Chung. Slowfast network for continuous sign language recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Patricia Cabot Álvarez, Xavier Giró Nieto, and Laia Tarrés Benet. Sign language translation based on transformers for the how2sign dataset. *Image Processing Group Signal Theory and Communications Department Universitat Politècnica de Catalunya. BARCELONATECH*.

J Bulas-Cruz, AT Ali, and Erik L Dagless. A temporal smoothing technique for real-time motion detection. In *International Conference on Computer Analysis of Images and Patterns*.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Multi-channel transformers for multi-articulatory sign language translation. In *European Conference on Computer Vision*.

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*.

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. 2022b. Two-stream network for sign language recognition and translation. In *2022 Neural Information Processing Systems*.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation . In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA. IEEE Computer Society.

Neil Cohn. 2013. *The Visual Language of Comics: Introduction to the Structure and Cognition of Sequential Images*.

Mathieu De Coster and Joni Dambre. 2022. Leveraging frozen pretrained written language models for neural sign language translation. *Information*, 13.

Mathieu De Coster, Karel D'Oosterlinck, Marija Pizurica, Paloma Rabaey, Severine Verlinden, Mieke Van Herreweghe, and Joni Dambre. 2021. Frozen pretrained transformers for neural sign language translation. In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, Virtual. Association for Machine Translation in the Americas.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota. Association for Computational Linguistics.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. 2021. How2sign: A large-scale multimodal dataset for continuous american sign language. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jay Gala, Pranjal A Chitale, A K Raghavan, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar M, Janki Atul Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, New York, NY, USA. Association for Computing Machinery.

Zifan Jiang, Gerard Sant, Amit Moryossef, Mathias Müller, Rico Sennrich, and Sarah Ebling. 2024. SignCLIP: Connecting text and sign language by contrastive learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, USA. Association for Computational Linguistics.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. ISLTranslate: Dataset for translating Indian Sign Language. In *Findings of the Association for Computational Linguistics: ACL 2023*, Toronto, Canada. Association for Computational Linguistics.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. CISLR: Corpus for Indian Sign Language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A benchmark for Indian Sign Language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, Bangkok, Thailand. Association for Computational Linguistics.

Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho. 2019. Neural sign language translation based on human keypoint estimation. *Applied Sciences*, 9.

Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, Barcelona, Spain. Association for Computational Linguistics.

Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. 2023. Gloss-free end-to-end sign language translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Toronto, Canada. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

MediaPipe. 2023. MediaPipe Holistic.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. Recent advances in natural language processing via large pretrained language models: A survey. *ACM Computing Surveys*, 56.

Amit Moryossef, Ioannis Tsochantaridis, Roee Aharoni, Sarah Ebling, and Srini Narayanan. 2020. Real-time sign language detection using human pose estimation. In *European Conference on Computer Vision*.

Amit Moryossef, Kayo Yin, Graham Neubig, and Yoav Goldberg. 2021. Data augmentation for sign language gloss translation. In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, Virtual. Association for Machine Translation in the Americas.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI*. Accessed: 2024-11-15.

Zed Sevcikova Sehyr, Naomi K. Caselli, Ariel Cohen-Goldberg, and Karen Emmorey. 2021. The asl-lex 2.0 project: A database of lexical and phonological properties for 2,723 signs in american sign language. *The Journal of Deaf Studies and Deaf Education*, 26.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany. Association for Computational Linguistics.

Samar Sinha. 2017. *Indian Sign Language: An Analysis of Its Grammar*. Gallaudet University Press.

Federico Tavella, Viktor Schlegel, Marta Romeo, Aphrodite Galata, and Angelo Cangelosi. 2022. WLASL-LEX: a dataset for recognising phonological properties in American Sign Language. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Dublin, Ireland. Association for Computational Linguistics.

Mindy Tran, Xinru Tang, Adryana Hutchinson, Adam J. Aviv, and Yixin Zou. 2024. Position paper: Exploring security and privacy needs of d/deaf individuals. In *2024 IEEE European Symposium on Security and Privacy Workshops (Euro S and PW)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online. Association for Computational Linguistics.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

Kayo Yin and Jesse Read. Attention is all you sign: sign language translation with transformers. In *Sign Language Recognition, Translation and Production (SLRTP) Workshop-Extended Abstracts*, volume 4.

Kayo Yin and Jesse Read. 2020. Better sign language translation with STMC-transformer. In *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. 2022. Spatial-temporal multi-cue network for sign language recognition and translation. *IEEE Transactions on Multimedia*, 24.

Appendix

Table of Contents

A Pose Stitched Dataset Details 10
A.1 Dataset created using linguistic Templates 10
A.2 Dataset Created using BPCC Corpus 10
A.3 Sentence Length Matching 10
A.4 Post-processing 10
A.5 Summary of Created Datasets . . 10
B Processing Stitched Poses 11
B.1 Framerate Matching 11
B.2 Pose Processing 12
C Hyperparameters/Training Details 13
C.1 Architecture Details 13
C.2 Linear Annealing 13
D Additional Results 13
D.1 Effect of Adding Pose-stitched Dataset 13
D.2 Effect of Distribution Shifts 14
D.3 Qualitative Analysis 14
D.4 Effects of random word order in the pretraining dataset. 14

List of Tables

2 Linguistic Phenomenon 11
3 Number of Sentences in Each Dataset. 11
4 Unique word counts ISL 12
5 Unique word counts ASL 12
6 Unique word counts ISL, ASL . . 12
7 Inspecting Pretraining effect on iSign dataset 15
8 Performance gain from Pretraining strategy 15
9 Set of hyperparameters used in the experiment. 15
10 Qualitative results on How2Sign . 16
11 Qualitative Results on BLIMP-ISL 16
12 ROUGE score result 16
13 Sacre score result 17

List of Figures

2 Data Creation Pipeline using different sources 11
3 Sentence length distribution ISL-Initial 17
4 Sentence length distribution ISL-final 17
5 Sentence length distribution ASL-Initial 17
6 Sentence length distribution ISL-Final 17
7 Sentence length distribution between How2Sign and iSign dataset 17
8 Vocabulary Distribution between CISLR, iSign train, BPCC-ISL . . 18
9 Vocabulary Distribution between WLASL, How2Sign, BPCC-ASL . 18
10 Vocabulary Distribution between BLIMP, CISLR, iSign train 18
11 Vocabulary Distribution between CISLR, BLIMP, BLIMP-ISL . . . 18
12 Vocabulary Distribution between WLASL, BLIMP, How2Sign . . . 18
13 Vocabulary Distribution between BLIMP-ASL WLASL, BLIMP . . 18
14 Frame Distribution between BPCC-ISL , iSign dataset 19
15 Frame Distribution between BLIMP-ISL , iSign dataset 19
16 Frame Distribution between BPCC-ASL , How2Sign dataset . 19
17 Frame Distribution between BLIMP-ASL , How2Sign dataset . 19
18 BLEU-4 with and without pose stitching 20
19 Validation BLEU scores on generated dataset 20
20 Data sampling over the training steps 20

A Pose Stitched Dataset Details

For the generated set of sentences, we found that the linguistic-template-based sentences cover a small set of words available in the corresponding sign language datasets (WLASL and CISLR). To explore further, we create another set using sentences from a translation dataset, BPCC (Gala et al., 2023). This helps increase the coverage of words, essentially improving the overlap with the target sign language datasets (How2Sign and iSign). Overall, we create four datasets using the proposed pose strategy: two using the linguistic templates from BLIMP (Warstadt et al., 2020), and two using the large corpus of translation text, BPCC (Gala et al., 2023).

Further, we perform additional preprocessing to match the generated sentence distribution with the target distribution to make the training more effective. In this section, we discuss the details of the entire dataset formation pipeline. An overview of the pipeline is represented in Fig. 2. In general, we have two target sign languages (ASL from How2Sign and ISL from iSign). We create the additional sentences using pose stitching to match the target distribution (both in terms of vocabulary as well as sequence length). We discuss each of the steps in the pipeline below:

A.1 Dataset created using linguistic Templates

We utilize the common vocabulary of BLIMP, CISLR, and WLASL for sentence generation. We generated the sentences by linguistic templates proposed in Benchmark of Linguistic Minimal Pairs (BLiMP) (Warstadt et al., 2020), which covers 67 linguistic paradigms with 12 phenomena. In total, we generated 22M sentences using the common vocabulary of BLIMP and CISLR. Similarly, using the common vocabulary of BLIMP and WLASL, we generated around 2.8M sentences. The dataset generated by linguistic templates using the common vocabulary of CISLR and BLIMP is referred to as BLIMP-ISL. The dataset generated by linguistic templates using the common vocabulary of WLASL and BLIMP is referred to as BLIMP-ASL. A sample of a few generated sentences is shown in Table 2 with their respective linguistic phenomenon.

A.2 Dataset Created using BPCC Corpus

The BPCC corpus (Gala et al., 2023) is a large dataset with 230 million bitext pairs. We matched the words in the sentences with the CISLR and WLASL vocabulary and selected the sentences with more than 90% word match. After selecting the sentences, we performed sentence merging(we merged small sentences) to match the length distribution of the sentences with the How2Sign and iSign datasets.

A.3 Sentence Length Matching

To match the sentence length distribution of the iSign dataset, we took 90% of the sentences from the dataset created using BPCC and CISLR with a length of less than 8, and merged three sentences into one. This merging and remaining 10% of the sentences resulted in 1.6M sentences, which we will refer to as the BPCC-ISL dataset. The sentence length distribution between the BPCC-ISL and iSign datasets before merging sentences is shown in Fig. 3, and after merging sentences is shown in Fig. 4. Similarly, different combinations of sentence merging on the dataset created by matching BPCC and WLASL resulted in 1.1 million sentences. We will refer to this dataset as BPCC-ASL. The sentence length distribution between BPCC-ASL and the How2Sign dataset before merging sentences is shown in Fig. 5, and after merging is shown in Fig. 6.

A.4 Post-processing

We further post-process the data by replacing the name of Person with <PERSON> and words with frequency less than 3 with <UNKNOWN> in the dataset (BPCC-ISL, BPCC-ASL) and from the train set of How2Sign and the train set of iSign.

A.5 Summary of Created Datasets

In total, we have two vocabulary data sets, CISLR and WLASL, four pose stitched datasets (BLIMP-ISL, BLIMP-ASL, BPCC-ISL, BPCC-ASL), How2Sign (train, test, val), iSign (train, test, val). The number of sentences in each dataset is shown in Table 3. Vocabulary count in iSign Train, CISLR, How2Sign, WLASL, BPCC-ISL, BPCC-ASL presented in Table 6 venn diagram of common vocab is shown in Fig.8, Fig.9. Vocabulary count and common vocab between CISLR, BLIMP, iSign, BLIMP-ISL is presented in Table 4 and a Venn diagram of common vocab is shown in Fig. 10 and Fig. 11. Vocabulary count and common vocab between WLASL, BLIMP, How2Sign, BPCC-ASL presented in Table 5 and a Venn diagram of common vocab is shown in Fig. 12 and Fig. 13.

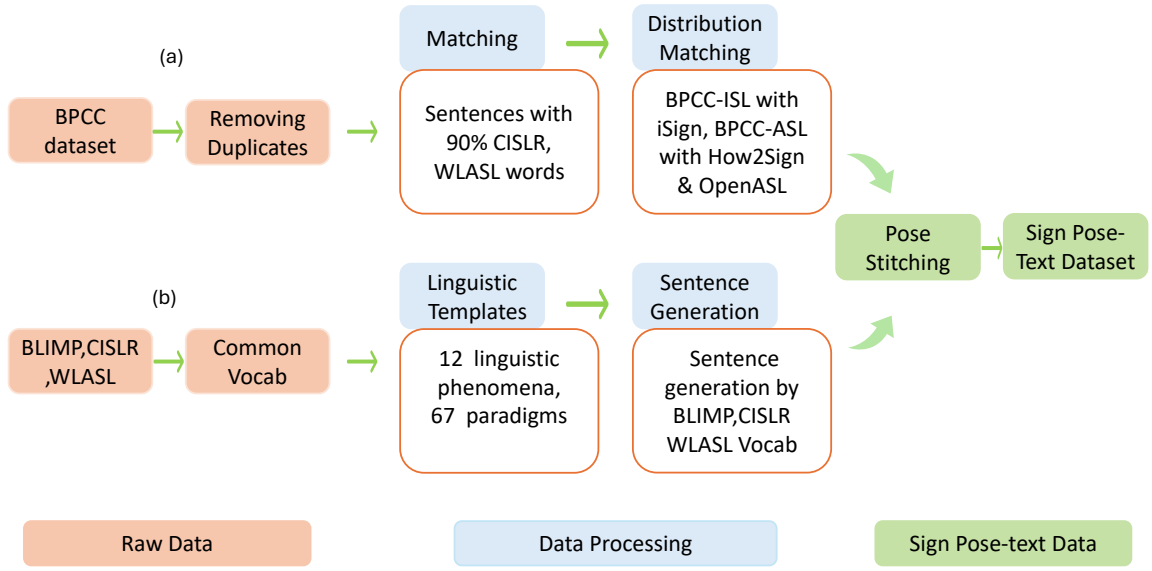

Figure 2: (a) From the filtered raw BPCC Corpus, sentences are matched with CISLR and WLASL vocabulary, and the sentence length distribution is aligned with iSign and How2Sign. Two datasets, BPCC-ISL and BPCC-ASL are created after pose stitching (b). Using templates from BLIMP (Warstadt et al., 2020) and the common vocabulary of BLIMP, CISLR, and WLASL, two additional datasets are created: BLIMP-ISL and BLIMP-ASL after pose stitching.

Phenomenon	N	BLIMP-ISL	BLIMP-ASL
ANAPHOR AGR.	2	*Some people will hide them- selves.*	*Many children will fire themselves.*
ARG. STRUCTURE	9	*Some boy should clean.*	*Some woman will stretch that jacket.*
BINDING	7	*some actress might think that she can wear that blouse.*	*It's herself that a girl can admire.*
CONTROL/RAISING	5	*that boy will predict it to be bad that all children wear this skirt.*	*that girl will find it to be bad that every grandmother can wear a coat.*
DET.-NOUN AGR.	8	*These children drink that beer.*	*that boy should wear this big coat.*
ELLIPSIS	2	*some boy will wear one big blouse and that girl will wear three.*	*some boy will wear one big coat and a girl will wear two.*
FILLER-GAP	7	*some boy will see some coat that people wear.*	*every boy can know that some boy should wear this coat.*
IRREGULAR FORMS	2	-	-
ISLAND EFFECTS	8	*Who should that father and some teacher visit?*	*who can boy's building this bank aid.*
NPI LICENSING	7	*Even people will often appear.*	*Even that cousin should also admire this pie.*
QUANTIFIERS	4	*that nephew can insult at least five children.*	*that brother should purchase at least six socks.*
SUBJECT-VERB AGR.	6	*The people hide these people.*	-

Table 2: Twelve Different Phenomenon from (Warstadt et al., 2020) and sentences from BLIMP-ISL and BLIMP-ASL dataset

Dataset Name	Number of Sentences
iSign Train	99,923
iSign Test	6,069
iSign val	5,653
How2Sign Train	31,092
How2Sign Test	2,349
How2Sign Val	1,739
BPCC-ISL	1,640,469
BPCC-ASL	1,173,536
BLIMP-ISL	22,219,407
BLIMP-ASL	2,880,008

Table 3: Number of Sentences in Each Dataset.

B Processing Stitched Poses

B.1 Framerate Matching

To match the frame rate of the two sources(i.e, for example BPCC-ISL and iSign), we made the mean of the two sources the same and sampled every xth frame. Where x denotes the sampling rate described in the C. Frame distribution between BPCC-ISL and iSign dataset before and after framerate matching is present in Fig. 14. Similarly, we perform a similar distribution matching on the How2Sign train and BPCC-ASL datasets. The frame distribution before and after the frame selection is present in Fig. 16 . Framerate matching also performed on linguistically generated dataset

Dataset Combination	Unique Words
iSign Train	15,093
CISLR	4,764
BLIMP-ISL	520
BLIMP	2,816
iSign Train ∩ CISLR	2,450
iSign Train ∩ BLIMP-ISL	457
iSign Train ∩ BLIMP	1,552
CISLR ∩ BLIMP-ISL	455
CISLR ∩ BLIMP	504
BLIMP-ISL∩ BLIMP	442
iSign Train ∩ CISLR ∩ BLIMP-ISL	439
iSign Train ∩ CISLR ∩ BLIMP	477
iSign Train ∩ BLIMP-ISL∩ BLIMP	426
CISLR ∩ BLIMP-ISL∩ BLIMP	435
All datasets	419

Table 4: Unique word counts in CISLR , BLIMP, iSign-Train, BLIMP-ISL

Dataset Combination	Unique Words
BLIMP-ASL	456
WLASL	2,000
BLIMP	2,816
How2Sign Train	7,430
BLIMP-ASL∩ WLASL	427
BLIMP-ASL∩ BLIMP	409
BLIMP-ASL∩ How2Sign Train	380
WLASL ∩ BLIMP	508
WLASL ∩ How2Sign Train	1,501
BLIMP ∩ How2Sign Train	1,084
BLIMP-ASL∩ WLASL ∩ BLIMP	405
BLIMP-ASL∩ WLASL ∩ How2Sign Train	366
BLIMP-ASL∩ BLIMP ∩ How2Sign Train	348
WLASL ∩ BLIMP ∩ How2Sign Train	433
WLASL ∩ BLIMP ∩ How2Sign Train ∩ BLIMP-ASL	344

Table 5: Unique word counts in WLASL , BLIMP, How2Sign-Train, BLIMP-ASL

BLIMP-ISL , BLIMP-ASLwith source dataset iSign and How2Sign. Frame Distribution in both cases is presented in Fig. 15, Fig. 17. On top of that, we randomly sampled the frames with frequency ranging from [1,3] to support generalization, as the videos were from various sources. Some were from the News channel ISH, and some were educational, making the speed of the signers vary significantly. We pick the selected keypoints from the selected frames and concatenate them to form a single vector of dimension 152 fed into our Encoder model.

B.2 Pose Processing

From the iSign and How2sign dataset videos, we extract the pose files using the media pipe library (MediaPipe, 2023). The media pipe library gives 576 key points as features. Sign language mainly involves manual(hand gestures) and non-manual (facial expression) markers. So, we focused on the upper body, especially the hands, and our facial expressions(eyebrows, lips, etc.). We hand-

Dataset Combination	Unique Words
iSign Train	15,093
CISLR	4,764
BPCC-ISL	6,700
iSign Train ∩ CISLR	2,450
iSign Train ∩ BPCC-ISL	4,976
CISLR ∩ BPCC-ISL	3,152
CISLR ∩ isign Train ∩ BPCC-ISL	2,449
WLASL	2,000
How2Sign Train	7,430
BPCC-ASL	4,613
WLASL ∩ How2Sign Train	1,501
WLASL ∩ BPCC-ASL	1,943
How2Sign Train ∩ BPCC-ASL	2,882
WLASL ∩ How2Sign Train ∩ BPCC-ASL	1,504

Table 6: Unique word counts for isign Train, CISLR, BPCC-ISL, How2Sign Train, WLASL, BPCC-ASL dataset and their overlaps

picked some key points inspired by (Lin et al., 2023), which play a key role in recognizing the sign appropriately. Mediapipe extracts 576 key points, out of which 21 are for the left hand, 21 are for the right hand, 33 are for the body pose, and 468 are for the face.

We took all the key points for hands as they are one of the most important modalities for sign language. Some of the pose landmarks were not as important for sign language translation, so we ignored them. We excluded the following pose keypoints: Rknee, Rankle, Rheel, Rfootindex, Lknee, Lankle, Lheel, Lfootindex, Leye(in), Leye(out), Reye(in), Reye(out), Mouth(2 keypoints), Lpinky, Rpinky, Lindex, Rindex, Lthumb, Rthumb, LHip, RHip. The respective keypoints are as follows: (26, 28, 30, 32, 25, 27, 29, 31, 1, 3, 4, 6, 9, 10, 17, 18, 19, 20, 21, 22, 23, 24). This leaves us with 11 key points out of the 33. For the face, we only took the following key points. Mouthright (61), Mouthleft (291), LipsLowerOuter (17), LipsUpperOuter(0), RightEyebrowUpper(70, 105, 107), LeftEyebrowUpper (300, 334, 336), RightEyeUpper (161,158), RightEyeLower (33, 163, 153, 133), LeftEyeUpper (388, 385), LeftEyeLower (263,390,380,362), Nosetop(9). A total of 23 key points for the face. We use a total of 76 key points, and each key point has a corresponding (x,y) coordinate, which makes it 152 key points per frame. Mediapipe gives the confidence of each key point that it predicts some of the key points were classified with very low confidence, we chose a threshold of 0.8, and if any key point was less than this threshold, then we filled the key points with the nearest

(left/right) frame of the video with the confidence of the keypoints surpassing the threshold.

C Hyperparameters/Training Details

Table 9 shows the hyperparameters used to train the architectures on different datasets. Some of the hyperparameters differ across the datasets, such as learning rate, number of encoder and decoder layers, attention heads, and dropout.

The sampling steps indicate the step at which we sample from the generated dataset. For example, a sampling rate of 3 indicates that we sample every 3rd frame in the generated dataset. The generated dataset has a larger number of frames as the frames are stitched together and are fetched from a vocabulary dataset, so the speed at which they are signed is also slow, effectively making the number of frames large in the generated dataset. This sampling rate is derived by making the mean of the frames of the dataset (iSign / How2Sign) similar to the mean of the frames of the generated dataset. So, if the mean of the number of frames of the generated dataset is x and the mean of the frames of the available dataset is y, then the sampling rate is equal to x/y.

For training, we follow two strategies: one where, while stitching the poses, the order of the words in the generated sentences is kept the same (same word order stitching (SWO)), and the second in which the order of the words in the generated sentences is shuffled randomly (random word order stitching (RWO)). We found this simple strategy of linearly moving towards the target distribution to work well across datasets.

C.1 Architecture Details

We follow an encoder-decoder model architecture for training an end-to-end SLT system. We took the BERT model (Devlin et al., 2019) as our encoder and the GPT2 model (Radford et al., 2019) as our decoder. We use the Huggingface transformers library (Wolf et al., 2020) for the implementation of the models. For the iSign dataset, both the encoder and the decoder had 4 layers with a hidden size of 512 with 8 attention heads each. We take the processed features/poses and feed them directly to our BERT model. For the decoder, we trained a BPE tokenizer (Sennrich et al., 2016) with our train data from both the generated data (BPCC-ISL, BLIMP-ISL) and the iSign dataset with a vocab of 15000. We took the ADAMW (Loshchilov and Hutter, 2019) as our optimizer with a learning rate of 3e-4 and a batch size of 16. We used 0.1 as our dropout. For a full set of hyperparameters, kindly refer to Table 9. For the pretraining strategy, we linearly increased the iSign data for training the model with a max threshold of 85 percent at 60000 steps. That is, we sample from a uniform random variable. If the sample is less than the threshold, we sample from iSign, or else we sample from the generated dataset. So, at the 0th step, we always sample from the generated dataset while linearly increasing the threshold to a maximum of 0.85. That is, after 60k steps, we sample from iSign 85 percent of the time. We follow a similar strategy for the How2Sign dataset with a few changes in hyperparameters. We followed two pretraining strategies. In the first strategy, we stitched the words/poses of the sentence in the same order, while in the second, the poses were stitched in random order. Table 1 shows the results for different pretraining.

C.2 Linear Annealing

During training, we follow a linear annealing strategy to first start from the constructed pose-stitched datasets and finally move towards the target distribution sentences from the respective sign language. More specifically, we linearly increase the number of samples that we take from the original dataset and reduce the samples that we take from the generated dataset. Fig. 20 shows the percent of data that we sample from the original dataset and the generated dataset as the number of training steps increases. Basically, we linearly increase the data from 0% (iSign/How2Sign) data at the 0th step to 85% of the (iSign/How2Sign) data at the 60000th step.

D Additional Results

D.1 Effect of Adding Pose-stitched Dataset

To further analyze whether training on the generated dataset provides any performance gain, we compared the performance of the system without training on the generated dataset to the performance after training on it. App. Table 8 shows a significant performance gain when the system is trained on the generated dataset, indicating that the proposed strategy is indeed helpful in improving the system's performance. We also ran an experiment with the same set of hyperparameters without adding any pose-stitched sentences to the training set. Fig. 18 highlights the performance boost obtained in both datasets.

Further, we also measure the performance of these models on new pose-stitched sentences not seen during training. Fig. 19 shows the obtained translation scores. We observe that the model does generalize over the created pose-stitched sentences, with a BLEU-4 score of 97 and 47 for different sets. We speculate that the primary reason for this boost is the enormous dataset size with less vocabulary, making it easier to learn a generalized representation. Moreover, the distribution of the generated sentences coming from the same set of vocabulary helps generalize better. A few sample translated pose-stitched sentences with ground truth sentences are shown in App. Table 11.

D.2 Effect of Distribution Shifts

To see if our model is learning the concepts present in the training data, we took the most similar and the least similar sentences of the train split to the test and val splits of the iSign dataset and computed the BLEU score on these. Similarly, we took the most similar and the least similar sentences from the generated dataset (BPCC-ISL) with the test and val splits of the dataset. We use SBERT embeddings to find the sentences with higher similarity and create two subsets with the highest and lowest similarity. Table 7 indicates a positive signal for the performance on the most similar sentences, while it performs worse on the least similar sentences, indicating that the model is able to learn the context. Moreover, this pattern repeats in the generated dataset, which indicates that if we increase the dataset such that the overlap with the original dataset is high, the performance of the system can significantly improve. Overall, we observe a performance improvement if the distribution of the sentences matches with the distribution of created pose-stitched sentences, showing the effectiveness of the proposed training strategy.

D.3 Qualitative Analysis

For the qualitative analysis, we included a few translated sentences and their corresponding ground truth sentences from the How2Sign dataset in Table 10. The Table shows the performance of our Model compared to GloFE. Our Model often captures the main idea better — for example, it correctly says “That’s a good question”, while GloFE gives unrelated or incorrect outputs. In some cases, like “It’s going to blend it in the hair”, our model is partly correct, showing better alignment with the reference. However, there are also hallucinations, like in “That’s called sympathetic magic”, our model repeats “magic magic magic”, which is incorrect. Overall, our model produces more relevant and accurate translations than GloFE.

D.4 Effects of random word order in the pretraining dataset.

Most sign languages have a distinct and often non-linear grammatical structure that differs significantly from English. However, due to the lack of accessible linguistic resources, reliable syntactic parsers, or annotated corpora for ASL or ISL grammar, we adopt English word order as a proxy, which simplifies generation and leverages the available textual infrastructure.

To explore how sensitive the model is to this assumption, we construct two variants of our synthetic datasets: **1)** ***Same Word Order (SWO):*** Poses are stitched in the same order as the English sentence, preserving syntactic structure and compositional cues. **2)** ***Random Word Order (RWO):*** Poses are stitched after randomly permuting the word order, injecting syntactic noise, and encouraging the model to learn flexible and robust representations. These two variants allow us to investigate the trade-off between syntactic alignment and generalization, especially in low-resource or cross-lingual sign language translation settings. For both How2Sign and iSign, we see that the models trained in the SWO fashion result in better performance. However, for How2Sign, RWO also comes close to the best model, indicating the model can robustly learn the representations.

Method	DEV				TEST			
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Most Similar Sentences (BPCC-ISL)	20.73	10.08	6.31	4.43	19.33	9.20	5.75	4.00
Least Similar Sentences (BPCC-ISL)	17.12	7.66	4.61	3.18	16.51	7.37	4.33	2.88
Most Similar Sentences(iSign train)	19.84	10.18	6.74	4.97	20.48	10.45	6.82	4.91
Least Similar Sentences(iSign train)	15.97	6.66	3.69	2.40	16.08	6.65	3.74	2.44

Table 7: Inspecting Pretraining effect on iSign dataset

Method	DEV				TEST			
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEU-1	BLEU-2	BLEU-3	BLEU-4
No PreTraining iSign	12.85	2.50	1.04	0.58	12.81	2.66	1.14	0.64
BPCC iSign (swo)	17.31	8.09	5.02	**3.54**	17.67	8.20	5.00	**3.43**
No Pretraining How2Sign	22.19	9.71	5.43	3.37	18.16	7.96	4.33	2.62
BLIMP How2Sign (swo)	26.89	13.36	7.92	**5.04**	25.98	12.55	7.25	**4.56**

Table 8: Performance gain from Pretraining strategy

Hyperparameter	Value(Isign, How2Sign)
Learning Rate	3e-4, 1e-4
Number of Encoder Layers	4, 2
Number of Decoder Layers	4, 2
Encoder Hidden Size	512
Decoder Hidden Size	512
Number of Attention Heads	8, 4
Dropout	0.1, 0.3
Max Frames (Truncate after these)	300
Number of Beams	3
Warmup Steps Ratio	0.1
Batch Size	16
LR Scheduler Type	constant_schedule_with_warmup
Max Length Decoder	128
Vocabulary Size Decoder	15000
Number of Keypoints	152
Weight Decay	0.01
Sampling Steps(x)	4, 3

Table 9: Set of hyperparameters used in the experiment.

Reference	GloFE	Our Model
Hi !	Hi!	Hi !
In this clip we are going to talk about dangers for these birds in the household and otherwise.	and i am going to show you what i want to do is you don't want it to do with it.	In this clip, we're going to talk about the bird
Well that's a good question.	it is not good for you.	That's a good question.
Cross over your arms to keep the bar steady and to hold there.	if you want to keep your feet and keep your tape.	It's very important to keep your arms straight and it's not much quicker.
That's called sympathetic magic.	you don't want to go too much.	It's called a magic magic magic.
Its really easy to use.	so, i'm going to show you how to do this.	It's easy to do.
Here we go.	it doesn't have any way.	So we go.
Take a deep breath in through your nose.	so, you're going to want to get a little bit of a little bit.	Inhale and exhale.
And you would paint it on your hair.	this is going to be the top of your head.	It's going to blend it in the hair.
And I'm going to take this one and extend it a little bit more.	the first thing you need to do is make sure you keep your hands.	So, you want to make a little bit more.
Can you swing your legs around?	you want to make sure that you get your feet.	You can go around the floor.

Table 10: Qualitative results on How2Sign comparing our model predictions and GloFE predictions with references. Red indicates hallucinated or incorrect segments, and blue indicates correct matches with the reference.

Reference	Prediction
Some people cure this student .	Some people cure this student .
Those children will dislike themselves .	Those children will dislike themselves .
This niece can buy this blouse .	This niece can buy this blouse .
People know that they climb down this ladder .	People know that they climb down this ladder .
Some truck might turn .	Some truck might turn .
These people teach that man .	These people teach that man .
That doctor should watch this daughter .	That daughter should watch this doctor .
This sweater will stretch .	This sweater will stretch .
Can that woman ever worry ?	That woman can worry .
That teacher can see that cheap hill .	That teacher can see that cheap hill .

Table 11: Reference vs Prediction sentence pairs generated using BLIMP template for qualitative comparison

	Methods	Source	DEV			TEST		
			ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
How2Sign(ASL)	GloFE-VN	–	15.48	3.40	13.36	15.39	3.52	13.27
	w/o Pose Stitched	–	17.76	4.38	14.85	16.55	3.67	13.98
	Pose Stitched (rwo)	BPCC	21.71	6.00	**18.28**	20.72	5.61	17.53
		BLIMP	20.55	5.86	17.34	19.90	5.23	16.83
	Pose Stitched (swo)	BPCC	21.27	5.67	18.04	**21.23**	**5.79**	**17.94**
		BLIMP	**21.36**	**6.11**	18.23	20.13	5.25	16.96
	Best	–	**21.36 (+5.88)**	**6.11 (+2.71)**	**18.28 (+4.92)**	**21.23 (+5.84)**	**5.79 (+2.27)**	**17.94 (+4.67)**
iSign (ISL)	GloFE-VN	–	10.26	1.32	9.26	9.72	1.32	8.88
	w/o Pose Stitched	–	9.23	0.53	8.25	1.49	0.40	1.45
	Pose Stitched (rwo)	BPCC	13.31	3.06	11.81	13.44	3.15	11.86
		BLIMP	11.15	1.87	09.95	11.72	2.06	10.41
	Pose Stitched (swo)	BPCC	**16.00**	**4.13**	**14.14**	**16.37**	**4.28**	**14.41**
		BLIMP	15.54	3.98	13.88	15.59	4.01	13.92
	Best	–	**16.00 (+5.74)**	**4.13 (+2.81)**	**14.14 (+4.88)**	**16.37 (+6.65)**	**4.28 (+2.96)**	**14.41 (+5.53)**

Table 12: ROUGE score results on the How2Sign and iSign datasets comparing different pretraining strategies (random word order vs. same word order), along with baseline results (GloFe). The numbers in brackets show the absolute improvements from the baseline.

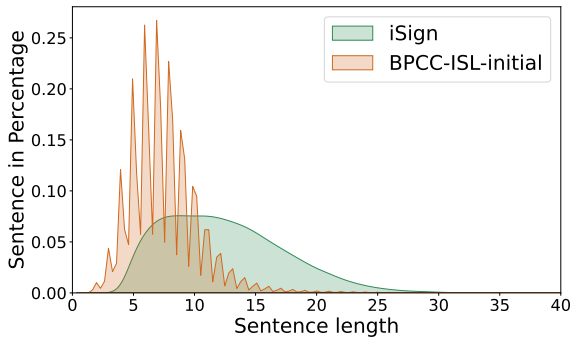

Figure 3: Sentence length distribution between iSign train and BPCC-ISL dataset before merging sentences

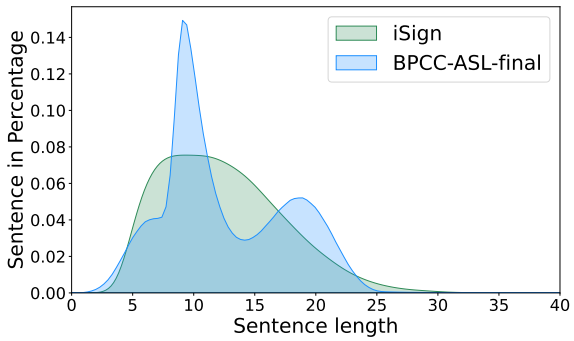

Figure 4: Sentence length distribution between iSign train and BPCC-ISL dataset after merging sentences

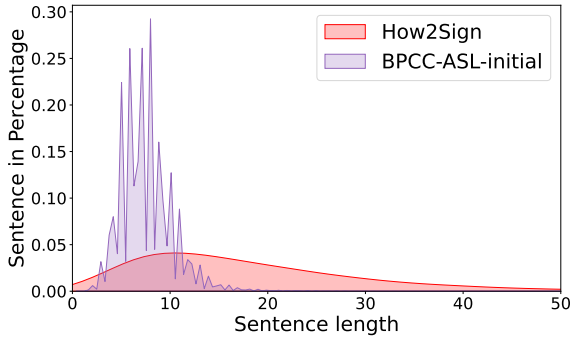

Figure 5: Sentence length distribution between How2Sign and BPCC-ASL dataset before merging sentences

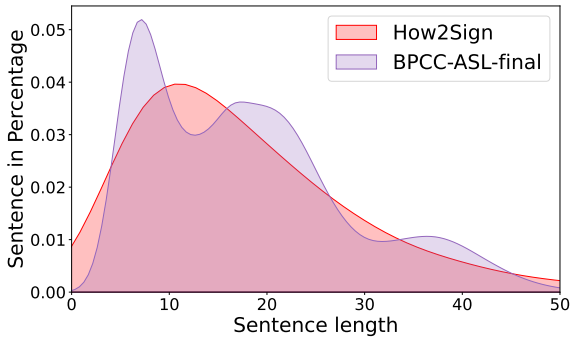

Figure 6: Sentence length distribution How2Sign and BPCC-ASL after merging sentences

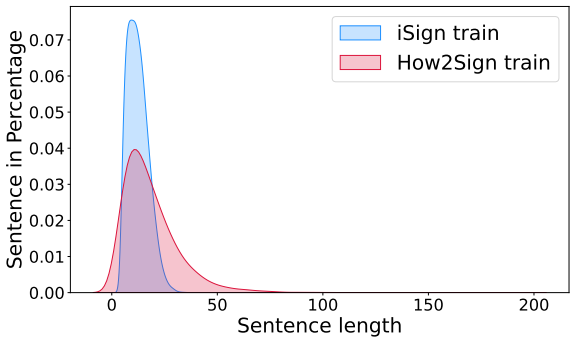

Figure 7: Sentence length distribution between How2Sign and iSign dataset

Dataset	Methods	Source	DEV				TEST			
			SacreBLEU1	SacreBLEU2	SacreBLEU3	SacreBLEU4	SacreBLEU1	SacreBLEU2	SacreBLEU3	SacreBLEU4
How2Sign	Pose Stitched (swo)	BLIMP	26.88	6.63	**2.79**	**1.29**	25.97	6.06	2.41	**1.13**
iSign	Pose Stitched (swo)	BLIMP	**21.99**	4.59	2.34	1.48	**22.36**	4.52	2.25	**1.37**

Table 13: Best SacreBLEU scores on the How2Sign and iSign datasets comparing different pretraining strategies (same word order).

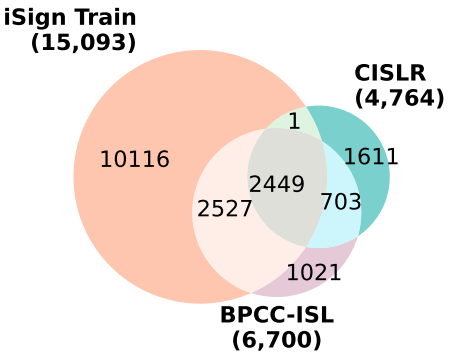

Figure 8: Vocabulary Distribution between CISLR, iSign train, BPCC-ISL

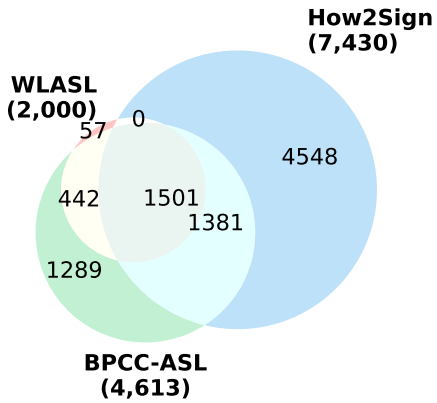

Figure 9: Vocabulary Distribution between WLASL, How2Sign, BPCC-ASL

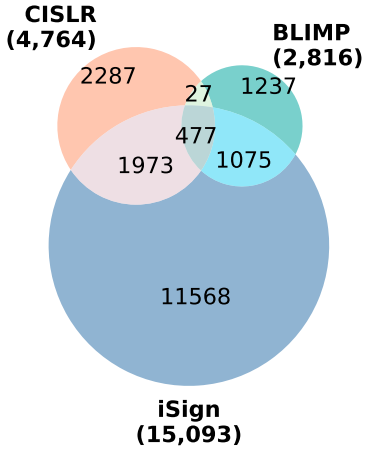

Figure 10: Vocabulary Distribution between BLIMP, CISLR, iSign train

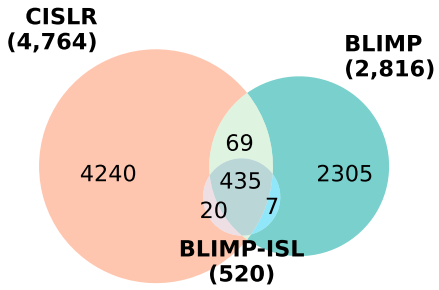

Figure 11: Vocabulary Distribution between CISLR, BLIMP, BLIMP-ISL

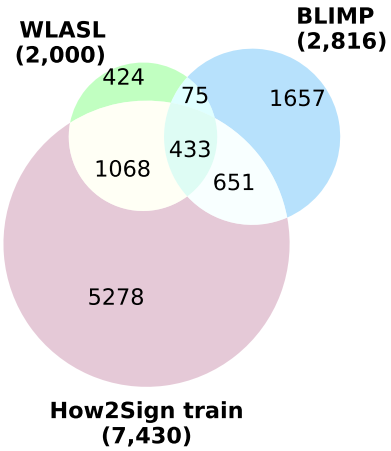

Figure 12: Vocabulary Distribution between WLASL, BLIMP, How2Sign

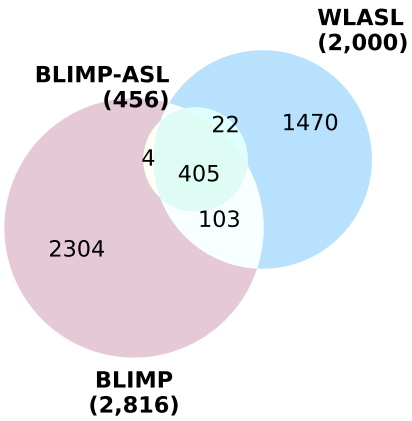

Figure 13: Vocabulary Distribution between BLIMP-ASL WLASL, BLIMP

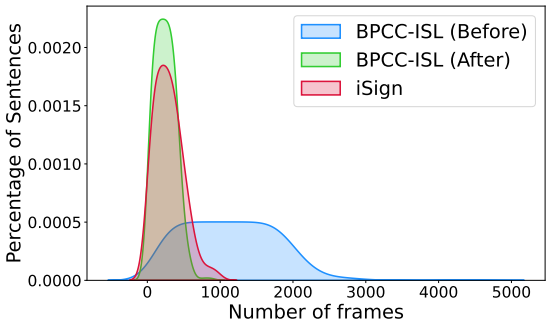

Figure 14: Frame Distribution between BPCC-ISL , iSign dataset

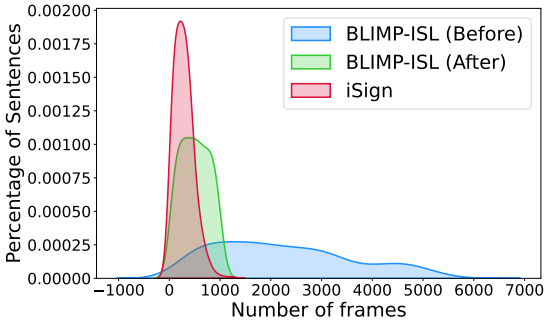

Figure 15: Frame Distribution between BLIMP-ISL , iSign dataset

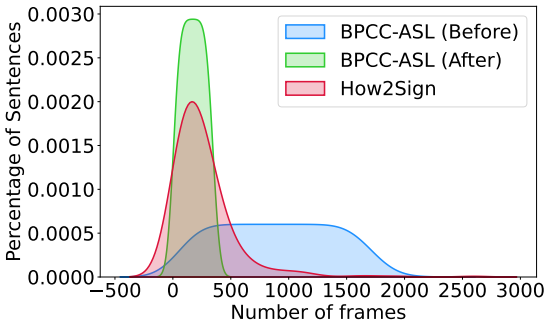

Figure 16: Frame Distribution between BPCC-ASL , How2Sign dataset

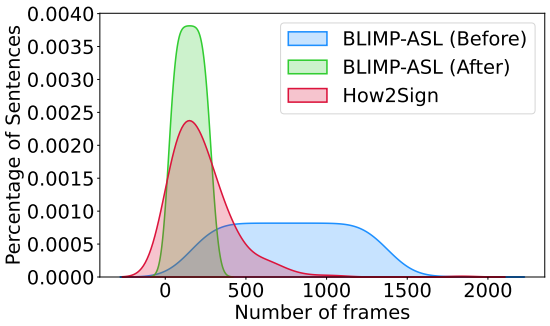

Figure 17: Frame Distribution between BLIMP-ASL , How2Sign dataset

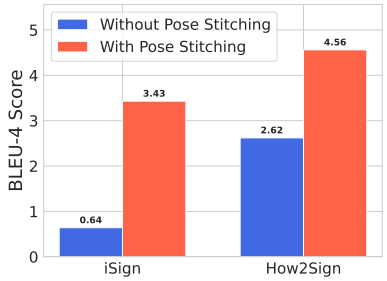

Figure 18: Performance gain from the proposed training strategy with added template-based generated sentences.

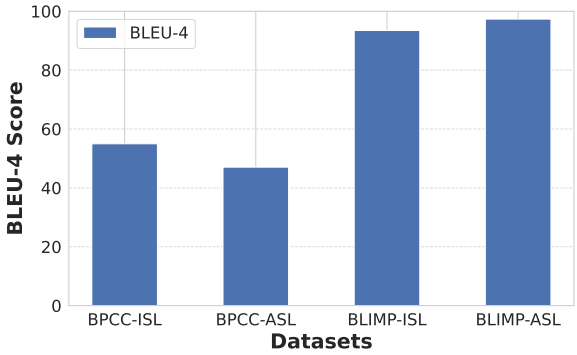

Figure 19: Validation BLEU scores on generated dataset

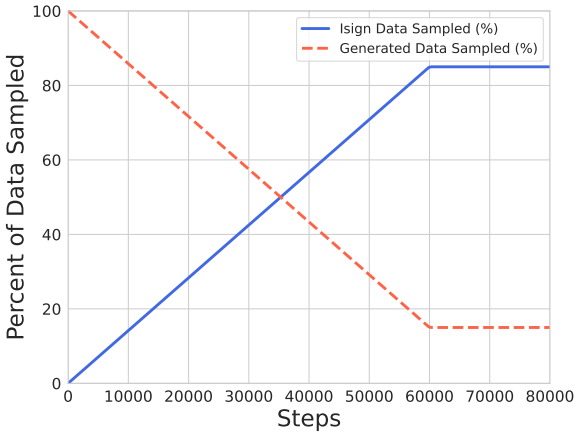

Figure 20: Data sampling over the training steps