BoostSLT: Boosting Sign Language Translation via a Plug-and-Play Diffusion-Based Semantic Enhancer

Supplementary Material

1. Additional Method Details

1.1. Energy-Aware Temporal Segmentation

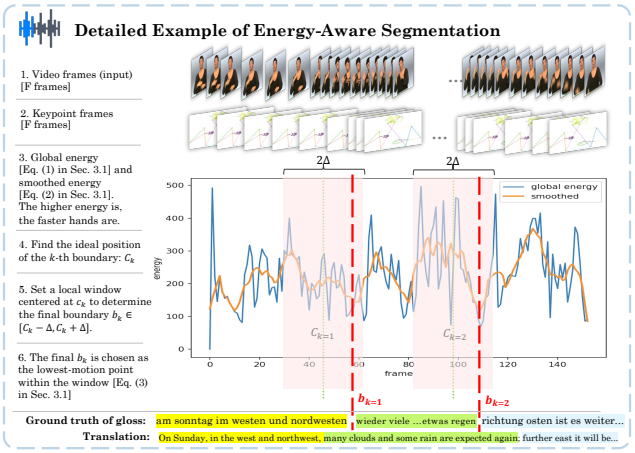

Figure 1. Details and example of EAT-Seg.

Temporal boundary modeling has been widely studied in long-horizon sequence understanding, where temporal convolutions and refinement modules are commonly used to capture long-range dependencies and suppress over-segmentation errors [1, 6, 9]. Here, we provide additional details of EAT-Seg to clarify how segment boundaries are determined. EAT-Seg computes hand-motion energy from hand keypoint dynamics, where lower-energy regions often correspond to natural pauses or transitions in signing. These local minima serve as candidate boundaries, as they are more likely to separate adjacent semantic units than high-motion regions.

To avoid unstable segmentation caused by signing-speed variation, boundary selection is not based solely on global minima. Instead, given the total number of frames and a target segment length, we first define a set of uniformly spaced centers c_k representing the expected boundary locations under a length-regularized partition. For each center, we search within a local temporal window and select the final boundary b_k by minimizing a joint cost combining motion energy and distance to c_k. This strategy balances semantic alignment and length stability, preventing overly short or overly long segments.

In addition, each segment is extended with a small temporal overlap. This overlap avoids cutting across semantically related signing motions and ensures complete coverage of gloss-relevant frames near the boundary. As a result, EAT-Seg produces segments that are both temporally stable and semantically coherent for downstream translation and reconstruction.

1.2. Diffusion-Based Semantic Reconstruction

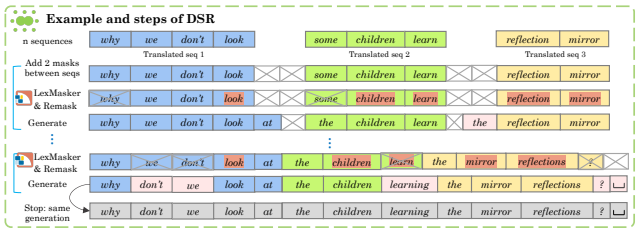

Figure 2. Details and examples of DSR.

DSR follows the general intuition of iterative masked refinement and discrete diffusion-style generation, which has shown strong potential for controllable and globally conditioned sequence generation [4, 7]. To further clarify the inference behavior of DSR, we describe here the complete reconstruction process. Given n translated segments, we first initialize a flexible sequence of length $L + 2(n-1)$ by inserting masked slots between adjacent segments, which provides the model with room for insertion, deletion, and local reorganization during denoising. Starting from this structured initialization, DSR performs iterative non-autoregressive refinement over the whole sequence rather than generating tokens from left to right.

During each refinement step, we employ a lightweight lexical guidance module, termed *LexMasker*, to control where re-masking should occur. LexMasker distinguishes content-bearing tokens, such as key nouns, verbs, numerals, and named entities, from low-information function words. Content tokens are preserved as semantic anchors whenever possible, while function words and unstable positions are preferentially re-masked and regenerated. This design allows DSR to improve sentence-level fluency and discourse structure without unnecessarily overwriting already-correct semantic content.

Inference is run for at most 20 diffusion steps. In practice, we terminate the process early when the masked positions remain unchanged for two consecutive iterations, which indicates that the reconstruction has reached a sta-

ble state. Unlike minimum Bayes risk decoding (MBRD), which only re-ranks a fixed set of hypotheses according to estimated utility [5], DSR actively rewrites the sequence through iterative denoising and therefore supports insertion, deletion, and reorganization. This enables it to correct long-range structural errors and missing or redundant content that are difficult to address through hypothesis selection alone.

2. Additional Empirical Analysis

2.1. Robustness to Segment-level Perturbations

BoostSLT[GASLT]	R-Del	R-Rep	R	B1	B2	B3	B4
✓	×	×	**46.72**	**45.67**	**33.05**	**26.54**	**21.95**
✓	×	✓	37.19	37.25	23.72	12.60	8.13
✓	✓	×	40.15	39.64	26.21	15.49	11.27

Table 1. Sensitivity of BoostSLT to artificially degraded segment-level translations for virtual upper-bound analysis.

To examine the dependency of BoostSLT on the quality of segment-level inputs, we conduct a perturbation analysis by artificially degrading the translated segments before reconstruction, which also provides a rough virtual upper-bound probe on how much BoostSLT could benefit from cleaner segment-level translations. Specifically, *R-Del* randomly deletes one token from each segment, while *R-Rep* randomly replaces one token in each segment. As shown in Table 1, performance drops under both perturbations, confirming that the final reconstruction quality is strongly influenced by the correctness of the intermediate segment translations. Nevertheless, the unperturbed setting remains substantially stronger, suggesting that BoostSLT can effectively exploit reliable segment-level evidence and may further benefit from higher-quality segmented translations.

2.2. Comparison with Alternative Semantic Enhancers

Auslan-Daily News	R	B1	B2	B3	B4
Diffusion SLT	15.14	16.53	4.92	2.01	0.62
BoostSLT[LiTFiC]	**39.98**	**38.45**	**23.29**	**16.63**	**13.57**

Table 2. Comparison with a directly trained text diffusion SLT model.

We additionally compare BoostSLT with an alternative semantic enhancement strategy based on direct text diffusion modeling [7, 8]. Specifically, we directly train a text diffusion model as a standalone SLT generator. As shown in Table 2, this direct diffusion SLT baseline performs substantially worse than BoostSLT. This suggests that under current data scale and visual–linguistic alignment constraints, diffusion models are more effective as reconstruction modules than as standalone end-to-end SLT generators.

2.3. Scalability and Error Propagation Analysis

Figure 3 shows that BoostSLT consistently improves SLT performance across backbone generations, although the gains are non-uniform. In general, weaker backbones with more severe long-range structural or coverage errors benefit more, while stronger backbones still obtain stable improvements. This trend highlights the scalability of BoostSLT: even if future large models become more capable of translating long unsegmented inputs directly, our plug-and-play framework remains attractive for efficient, deployable, and on-device SLT enhancement.

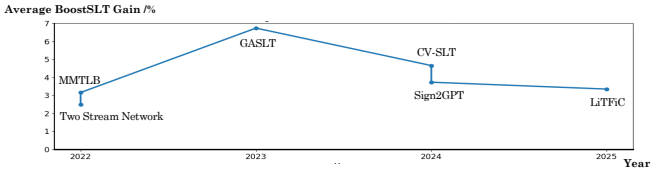

Figure 3. Trend of Average Performance Gains by BoostSLT

Position of errors	1–5	5–10	10–15	15+
TwoStreamNetwork	0.23	0.31	0.49	0.61
BoostSLT[TwoStreamNetwork]	0.20	0.24	0.27	0.32

Table 3. Error rate at different word positions.

Autoregressive sequence generation is known to suffer from exposure bias and cumulative error propagation, where early mistakes may propagate to later positions during inference [2, 3]. To examine whether BoostSLT mitigates this effect, we report the average error rate across different sentence positions on PHOENIX-2014T. As shown in Table 3, the baseline TwoStream Network exhibits increasing errors toward later positions, whereas BoostSLT substantially reduces errors in the 10–15 and 15+ ranges. These results provide further evidence that diffusion-based reconstruction suppresses error accumulation and improves discourse-level consistency over longer outputs.

2.4. Failure Analysis

Figure 4. Representative failure cases.

We further provide representative failure cases to illustrate the current limitations of BoostSLT. Although DSR is effective at improving sentence-level fluency and repairing local inconsistencies, its final output still depends on the quality of the translated segments produced upstream. When the intermediate segments already contain substantial semantic mismatch or miss critical content words, the reconstruction stage may be guided toward a fluent yet incorrect sentence. These cases suggest that future improvements may come from stronger segment-level translation quality, better uncertainty-aware masking, and tighter interaction between segmentation and reconstruction.

References

[1] Yazan Abu Farha and Juergen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Pro-*

ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3575–3584, 2019. 1

[2] Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Chi Kit Cheung. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 700–710, 2022. 2

[3] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, 2015. 2

[4] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022. 1

[5] Markus Freitag, David Grangier, Qijun Tan, and Bowen Liang. Minimum bayes risk decoding with neural metrics of translation quality. *arXiv preprint arXiv:2111.09388*, 2021. 2

[6] Colin Lea, Michael D. Flynn, René Vidal, Austin Reiter, and Gregory D. Hager. Temporal convolutional networks for action segmentation and detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017. 1

[7] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Tatsunori B. Hashimoto, Percy Liang, Dan Jurafsky, and Daniel Fried. Diffusion-lm improves controllable text generation. In *Advances in Neural Information Processing Systems*, 2022. 1, 2

[8] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025. 2

[9] Fangqiu Yi, Hongyu Wen, and Tingting Jiang. Asformer: Transformer for action segmentation. In *Proceedings of the British Machine Vision Conference*, 2021. 1