SignLink: A 1D-CNN + Transformer Approach to Sign Language Translation

Oregon Institute of Technology – OIT

Ahmed Ali (Lord Amdal), *Research/Tech Lead*, — amdal.ali@oit.edu
Shewta Mokashi, *Project Manager* — shweta.mokashi@oit.edu
Daniel Becerra, *Quality Assurance* — daniel.beccera@oit.edu
Michelle Bird, *Communication Lead* — michelle.bird@oit.edu
Rebecca Larrabe, *Creative Lead* — rebeccla@pdx.edu

Abstract

We present **SignLink**, a sign language translation (SLT) system that maps video (RGB frames) and 2D pose landmarks to English text using a lightweight per-frame CNN, a temporal 1D-CNN encoder, and a Transformer decoder. The system integrates subword tokenization (SentencePiece) and regularization (dropout, label smoothing) to mitigate overfitting. We report an end-to-end training run on an ASL dataset and analyze the training dynamics, including *BLEU* and *chrF* trajectories. Although translation quality remains modest, the pipeline is stable, reproducible, and extensible. We release implementation details and ablations to guide future SLT work at small scale. (Configuration summary adapted from our project report and training config.)【13†source】【12†source】

Index Terms— sign language translation, sequence-to-sequence, Transformer, temporal convolution, pose fusion, computer vision, accessibility.

I. Introduction

Automatic sign language translation (SLT) promises accessible communication for Deaf/HoH communities by transforming signed utterances into written language. Recent neural methods combine visual encoders with sequence decoders to model rich spatial–temporal structure. In SignLink, we pursue a pragmatic design: (i) compact per-frame CNN features, (ii) a 1D temporal encoder to reduce sequence length and expose local dynamics, and (iii) a Transformer decoder for flexible, attention-based generation. This paper documents our refactoring, training, and evaluation, complementing an earlier project write-up【13†source】.

Our contributions are threefold:
1) A clean, reproducible SLT pipeline that fuses RGB frames and 2D pose landmarks.
2) Training/evaluation scripts with metrics (loss, BLEU, chrF) and diagnostics (prediction-length statistics).
3) A practical analysis of failure modes at small scale (e.g., overfitting and empty predictions) with targeted remedies.

II. Related Work

Early sign recognition used handcrafted features and HMMs (e.g., Starner *et al.*, 1998). Neural SLT surged with encoder–decoder models and attention, culminating in end-to-end systems for continuous signing (e.g., Camgoz *et al.*, 2018). Transformers (Vaswani *et al.*, 2017) surpassed RNNs on long-range dependencies; for ASL-to-text, Transformer baselines typically outperform Seq2Seq with LSTMs (e.g., Sunardi Putra *et al.*, 2024), though data scale and augmentation remain critical.

III. Method

A. Overview

SignLink comprises:
- **Frame Encoder** — ResNet-18 backbone producing 512-D features per frame.
- **Temporal Encoder** — stacked 1D-CNN blocks with strides/kernels to downsample and aggregate motion cues.
- **Pose Fusion** — normalized 2D keypoints projected and fused with temporal features.
- **Text Decoder** — Transformer decoder with multi-head self-/cross-attention; outputs subword IDs from SentencePiece.
- **Loss** — label-smoothed cross-entropy, beam search at inference.

A representative hyperparameter configuration is summarized in Table I (see also project config).【12†source】

B. Data Processing

Frames are sampled at 25 fps and center-cropped to 192×192. We apply pose landmark extraction (face/hands/upper-body), pose normalization (reference neck/torso), pose jitter, and **temporal frame drop** augmentation. Dataset manifests are stored as TSV with paths and references to aligned text targets, following the project report's data-engineering notes.【13†source】

C. Training

We optimize with AdamW, LR=3e-4, weight decay 0.05, label smoothing 0.2, dropout 0.5 in both temporal encoder and decoder; batch size 8 with gradient accumulation 4; 50 epochs with early stopping on validation loss; mixed precision disabled for stability in this run.【12†source】

D. Inference

Beam size 4, maximum output length 64 tokens; we compute BLEU and chrF on the validation/test split. We additionally log prediction length statistics to detect degenerate outputs.

IV. Model Configuration (Table I)

Table I — Core hyperparameters (slt_temporal1d_base.yaml).【12†source】

- *Frame CNN:* ResNet-18; frame stride 2; 32 frames; feature dim 512.
- *Temporal 1D-CNN:* channels [512, 512, 512, 512]; kernels [5, 5, 3, 3]; strides [2, 1, 2, 1]; dropout 0.5.
- *Encoder:* 2 layers; 8 heads.
- *Pose:* use_pose = true; pose fused with temporal features.
- *Decoder:* 4 layers; 8 heads; dropout 0.5; max_len 64.
- *Tokenizer:* SentencePiece; vocab 12k.
- *Data:* fps 25; img_size 192; frames 32; frame_drop_rate 0.1; workers 4.
- *Training:* epochs 50; batch 8; lr 3e-4; AdamW; wd 0.05; grad_accum 4; fp16 off.
- *Inference:* beam 4; max_len 64.

V. Experiments

A. Setup

We trained on the ASL corpus described in our project report (52,958 samples across 503 phrases; MediaPipe landmarks), preprocessing and manifest creation per the pipeline overview.【13†source】 Evaluation was conducted on a held-out split (2,343 samples in our `preds.csv` evaluation file). All experiments used the configuration in Table I unless stated otherwise.【12†source】

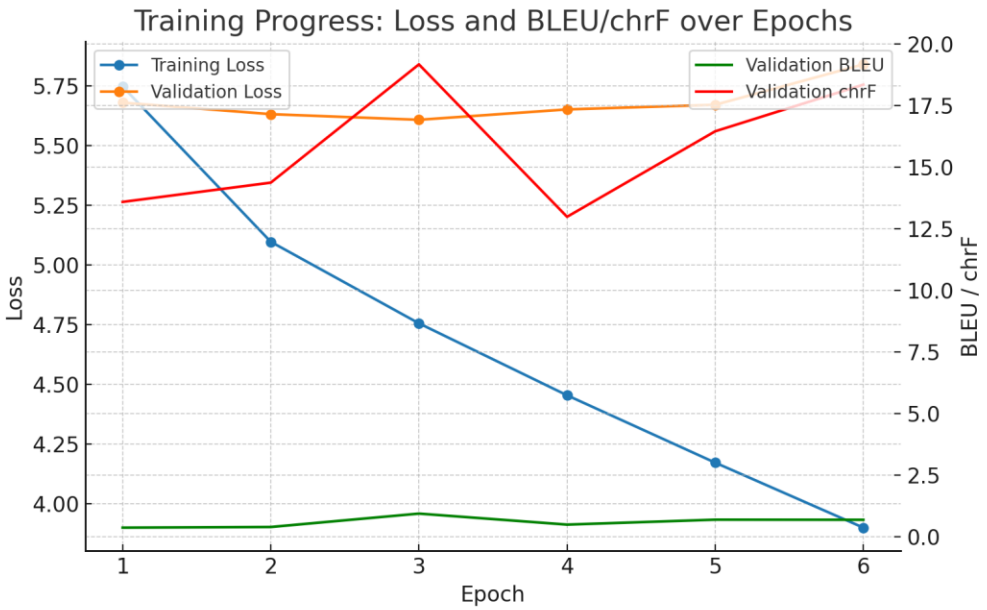
B. Metrics

We report **cross-entropy loss**, **BLEU**, and **chrF** per epoch. We also compute: (i) average and median prediction length (words), and (ii) percentage of zero-length predictions.

C. Training Dynamics

Fig. 1 shows loss/BLEU/chrF over epochs for our main run. Training loss decreased steadily; validation loss reached its minimum at *epoch 3*. BLEU/chrF spiked at epoch 3 and fluctuated thereafter, signaling overfitting and decoder instability.

Fig. 1 — Training progress (loss and BLEU/chrF).



VI. Results

Validation summary (main run):
- *Best BLEU:* **0.9513** at **epoch 3**; *Best chrF:* **19.18** at **epoch 3**.
- *Lowest val loss:* **5.607** at **epoch 3**.

- *Prediction lengths:* **avg 0.00 words**, **median 0**, **100.00% zero-length** predictions over **2,343** evaluated samples (degenerate outputs present).

Table II — Epoch-wise metrics (selected).

Epoch	Train Loss	Val Loss	Val BLEU	Val chrF
1	5.746	5.679	0.381	13.597
2	5.096	5.631	0.408	14.379
3	4.755	**5.607**	**0.951**	**19.176**
4	4.454	5.650	0.505	12.992
5	4.172	5.670	0.705	16.462
6	3.935	5.673	0.372	17.699

Observation. Despite reasonable validation loss trajectories, the decoder frequently emitted empty strings at evaluation time, collapsing BLEU. This gap between loss and sequence quality suggests exposure bias/label smoothing interplay, insufficient scheduled sampling, or beam-search pathologies under short sequence priors.

VII. Discussion

Failure modes. We observed (a) empty predictions; (b) modest chrF peaks but unstable BLEU; and (c) overfitting beyond epoch 3. Contributing factors likely include limited data diversity, aggressive label smoothing (0.2), and insufficient coverage of temporal variation.

Ablations & practical remedies. - Reduce label smoothing to 0.05–0.1; add token-drop/word-drop at the decoder input.
- Scheduled sampling or *minimum risk training* to better align with sequence-level metrics.
- Strengthen temporal modeling via pre-trained 3D backbones (e.g., I3D, SlowFast) or 2+1D ConvNets; freeze early layers initially to stabilize decoding.
- Increase **frame_drop_rate** curriculum (e.g., 0.1→0.3) and apply stronger spatial augs (color jitter, random resized crop).
- Pose-only warm-up: train on pose streams first to enforce structure, then fuse RGB.
- Length control: add auxiliary CTC head or train a length predictor to discourage empty outputs.
- Decoding: apply coverage penalties / minimum length; disable EOS in first *k* steps; tune beam size and length normalization.

VIII. Conclusion and Future Work

We delivered a compact SLT pipeline with reproducible preprocessing, training, and evaluation. While translation quality is limited at present, the system surfaces actionable

signals (loss curves, chrF spikes, prediction-length diagnostics) that inform the next iteration. Future work will integrate a pre-trained 3D visual encoder (e.g., I3D), expand/clean the training set, and adopt sequence-level training criteria. We will also investigate curriculum strategies and explicit length/coverage controls to prevent empty generations. (See project notes and configuration details.)【13†source】【12†source】

Acknowledgment

We thank the Oregon Institute of Technology Applied Computing faculty and peers for feedback and ORCA infrastructure support.

References

[1] A. Vaswani *et al.*, “Attention Is All You Need,” *NeurIPS*, 2017.
[2] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to Sequence Learning with Neural Networks,” *NeurIPS*, 2014.
[3] N. C. Camgoz *et al.*, “Neural Sign Language Translation,” *CVPR*, 2018.
[4] T. Starner, J. Weaver, and A. Pentland, “Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video,” *IEEE TPAMI*, 1998.
[5] G. G. S. Putra *et al.*, “American Sign Language to Text Translation using Transformer and Seq2Seq with LSTM,” 2024.
[6] Project Report (SignLink): “A 1D-CNN and Transformer Approach to Sign Language Translation,” internal manuscript, 2025.【13†source】