REFINESLT: LEARNING WHAT TO RECOVER AND WHEN TO REFINE IN SIGN LANGUAGE TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing sign language translation (SLT) models can produce fluent sentences while omitting or mistranslating content essential to the signed message, such as entities, actions, or temporal information. Addressing these errors requires identifying potentially missing or mistranslated content and directing refinement toward its visual evidence. We introduce RefineSLT, a framework that couples semantic cue prediction with learned refinement decisions for pretrained SLT models. CueMiner derives semantic cue supervision from training translations, and CueGrounder learns to predict these cues and associate them with temporal visual evidence. CueRefiner combines cue confidence with estimated coverage in the initial translation to control bounded, temporally weighted feature updates. The pretrained decoder then generates the final translation from the refined features. The backbone encoder and decoder remain frozen, and inference requires neither reference translations nor ground-truth glosses. With enhancement modules trained for each backbone, the framework is designed to accommodate different architectures, translation capabilities, and sign languages, offering a common basis for enhancing existing translators and adapting to future models. An analysis of baseline outputs on PHOENIX-2014T illustrates how errors in a few content-bearing words or phrases can alter the intended message, motivating semantic cue learning and selective refinement. Experiments on PHOENIX-2014T, How2Sign, and Auslan-Daily demonstrate consistent BLEU-4 and ROUGE-L improvements across all evaluated backbones, including relative BLEU-4 gains of 12.1% on PHOENIX-2014T and 54.2% on Auslan-Daily Communication with CV-SLT.

1 INTRODUCTION

Sign language translation (SLT) aims to convert continuous visual–gestural language into spoken-language text. This requires learning correspondences between languages with different grammatical structures and modes of expression from limited paired data. Existing approaches broadly follow gloss-based or gloss-free paradigms. Gloss-based methods exploit intermediate sign annotations to support recognition and translation (Camgoz et al., 2018; 2020; Chen et al., 2022b), whereas gloss-free methods learn from video–text pairs without manual gloss supervision (Yin et al., 2023; Gong et al., 2024; Wong et al., 2024). Recent advances in visual–text alignment and language-model-assisted generation have further improved the modeling of visual and linguistic context (Jiao et al., 2024; Jang et al., 2025).

Central to SLT is semantic fidelity: preserving the message expressed by the signer. An important consideration is that a small change in wording can have a substantial effect on sentence meaning. Evidence from NLP illustrates this sensitivity from complementary perspectives. Contrast-set evaluations show that small, targeted edits can change the correct interpretation of an input (Gardner et al., 2020); CheckList exposes failures in handling distinctions involving negation, temporal expressions, and semantic roles (Ribeiro et al., 2020); and PAWS demonstrates that sentences with nearly identical words can express different meanings when the relations between those words change (Zhang et al., 2019). Together, these findings motivate attention to meaning-critical words and phrases in their sentence context. In SLT, omitting or mistranslating even a small number of meaning-critical words or phrases can distort the signed message, even when the output remains fluent. For example, translating low pressure as high pressure changes the central claim of an otherwise plausible weather

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

description. Faithful translation therefore requires preserving these words or phrases together with their contextual meaning.

Existing SLT methods address aspects of this requirement through semantic supervision and richer linguistic context. GASLT encourages attention within locally coherent semantic regions and transfers sentence-level similarity knowledge (Yin et al., 2023), while VAP learns visual–text correspondences to improve visual representations (Jiao et al., 2024). GloFE uses text-derived concepts as weak supervision for visual representation learning (Lin et al., 2023). Contextual translation further enriches generation with background descriptions, preceding translations, and pseudo-glosses (Jang et al., 2025). These advances motivate a complementary question: how can semantic evidence be combined with an estimate of what an initial translation already expresses to guide targeted feature refinement? To examine this issue in existing translators, Figure 1 compares GASLT and TwoStreamNetwork outputs with reference translations on PHOENIX-2014T, combining a distribution of keyword-error counts with qualitative examples. The examples show that errors in a few content-bearing words or phrases can alter the central message despite substantial overlap with the reference: incorrect temperatures change the forecast, while substituting north for east changes the wind direction. The distribution summarizes the frequency of the counted keyword errors, and the examples illustrate their semantic consequences. Together, these observations motivate targeted recovery of meaning-critical content, guided by visual evidence and an estimate of what the initial translation already expresses.

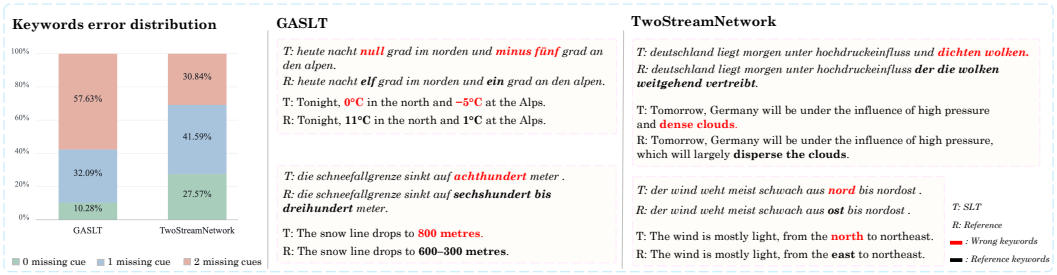


Figure 1: Keyword-level errors and their semantic impact on PHOENIX-2014T. Left: Distribution of missing reference-derived semantic cues in GASLT and TwoStreamNetwork outputs. Up to two cues are selected per sentence. A cue is counted as missing if it does not appear as a contiguous token sequence after lowercasing, German umlaut normalization, and punctuation removal. "2 missing cues" means both selected cues are absent. Right: Qualitative examples of baseline translations (T) paired with references (R), with English translations for readability. Red text marks incorrect baseline content; bold reference text highlights its counterpart. Localized errors can alter temperatures, cloud conditions, or wind direction despite otherwise similar wording.

Addressing these errors without reference translations at inference presents three challenges: **(1) Semantic definition:** How can we define and identify words or phrases whose omission or mistranslation substantially changes the intended meaning? **(2) Evidence localization:** How can we infer these keywords and locate their corresponding visual evidence when reference translations are unavailable at inference? **(3) Targeted refinement:** How can we improve the translation of these keywords while preserving content that is already correct?

To address these challenges, we introduce REFINESLT, a framework for refining pretrained sign language translators through semantic cues. We define a semantic cue as a context-dependent word or phrase whose omission or substitution substantially changes the central meaning of a sentence. Three components connect this definition to translation refinement. First, *CueMiner* derives cue pseudo-labels from training translations by measuring the change in sentence embeddings caused by masking candidate spans, complemented by an IDF prior. This provides an operational measure of semantic importance without manual cue annotation. Second, *CueGrounder* learns to predict cue representations from temporal visual features and the decoder states of an initial translation, and associates each cue with supporting video evidence through weakly supervised temporal alignment. Access to visual features allows cue prediction to extend beyond content already expressed in the initial translation. Third, *CueRefiner* estimates cue coverage in the initial decoder states and combines it with prediction confidence to control bounded, temporally weighted residual updates. This

directs refinement toward confident cues with low estimated coverage while suppressing updates for uncertain or already represented content.

CueMiner operates only during training. At inference, CueGrounder and CueRefiner use the video features and the backbone's own initial decoding states, after which the original decoder generates the final translation from the refined features. The backbone remains frozen, and the refinement modules are trained for each selected architecture. RefineSLT thus learns to use semantic evidence retained in an existing translator to guide both the location and strength of feature refinement. Its selectivity follows from estimated cue confidence and coverage, without assuming that every predicted cue requires a correction.

Our contributions are threefold:

- We formulate semantic refinement of pretrained SLT models around context-dependent semantic cues, linking what content to recover, where its visual evidence occurs, and how strongly to refine the corresponding features.
- We develop CueMiner and CueGrounder to convert contextual semantic importance into training supervision and learn video-conditioned cue prediction with temporal grounding, without manual cue or temporal annotations.
- We introduce CueRefiner, which combines cue confidence and estimated translation coverage to control localized residual updates, enabling selective refinement through the original frozen decoder.

2 Related Work

2.1 Sign Language Translation

Sign language translation (SLT) broadly follows gloss-based and gloss-free paradigms. Gloss-based methods use glosses as intermediate representations or auxiliary supervision (Camgoz et al., 2018; 2020; Chen et al., 2022b), whereas gloss-free methods learn from video–text pairs without manual gloss annotations (Yin et al., 2023; Gong et al., 2024; Wong et al., 2024). Complementary directions address data scarcity through sign back-translation (Zhou et al., 2021) and enrich visual representations with multiple articulators (Zhou et al., 2022) or facial expressions (Zheng et al., 2021). Recent advances strengthen semantic alignment and linguistic context. GASLT (Yin et al., 2023) combines locally coherent gloss attention with sentence-level similarity transfer, while VAP (Jiao et al., 2024) derives visual–text alignment constraints from translation sentences. Language-model-based approaches exploit pretrained linguistic knowledge (Gong et al., 2024; Wong et al., 2024; Liu et al., 2025), and contextual translation incorporates background descriptions, preceding translations, and pseudo-glosses (Jang et al., 2025). RefineSLT complements these advances by learning cue-guided feature refinement for a selected pretrained backbone while keeping its encoder and decoder frozen.

2.2 Semantic Supervision and Temporal Grounding

Text-derived semantic supervision connects our approach to keyphrase extraction and concept-based visual learning. EmbedRank (Bennani-Smires et al., 2018) uses sentence embeddings to select semantically relevant, diverse keyphrases. MDERank (Zhang et al., 2022) ranks candidate keyphrases by comparing document embeddings before and after masking. CueMiner uses a related masking-based criterion at the sentence level, combined with an IDF prior and redundancy suppression, to construct training-only cue pseudo-labels. In SLT, GloFE (Lin et al., 2023) uses text-derived concepts to query visual features and learns concept-related representations through contrastive supervision. RefineSLT shares this use of textual semantics as supervision and learns to predict context-dependent cues from video features and initial decoder states.

Locating the corresponding visual evidence is related to sequence alignment and sign spotting. In gloss-supervised recognition, iterative alignment (Pu et al., 2019) and visual alignment constraints (Min et al., 2021) strengthen the correspondence between visual sequences and gloss labels. Watch, Read and Lookup (Momeni et al., 2020) integrates sparse sign labels, subtitles, and dictionary examples through multiple-instance learning and noise-contrastive estimation. Bull et al. (Bull et al., 2021) study subtitle alignment using manually annotated temporal correspondences, while

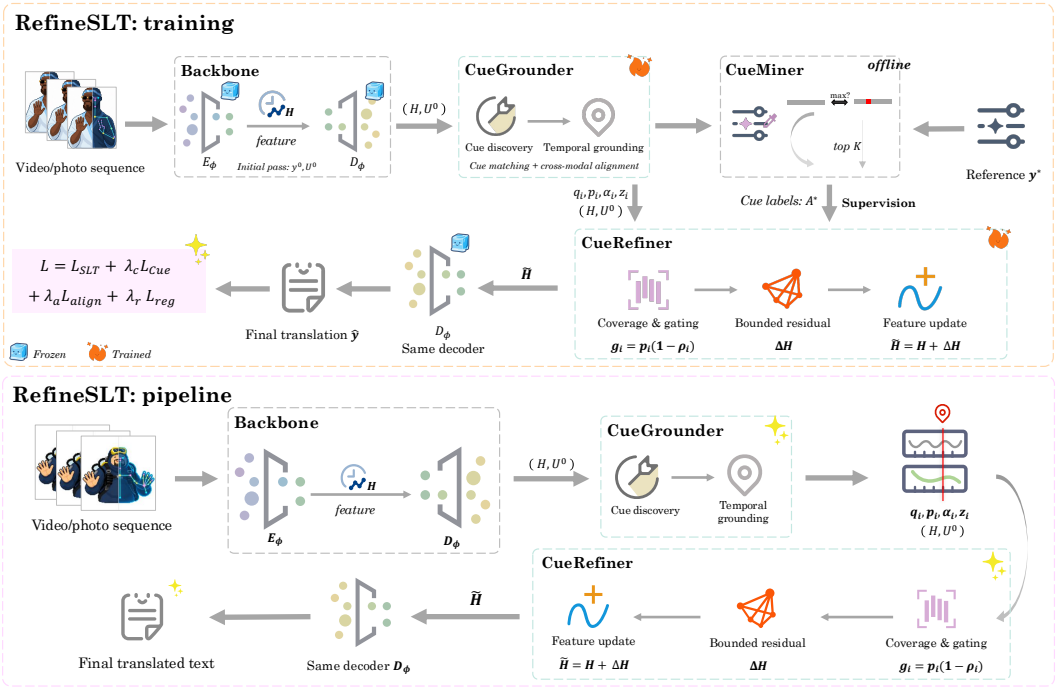


Figure 2: Overview of REFINESLT.

CiCo (Cheng et al., 2023) learns fine-grained sign–word mappings for cross-lingual video–text retrieval. CueGrounder learns soft temporal associations from sentence-level cue supervision without annotated temporal boundaries. Its grounded evidence is used to guide translation feature updates.

2.3 COVERAGE MODELING AND TRANSLATION REFINEMENT

Coverage modeling and multi-pass generation provide related mechanisms for addressing translation errors. Tu et al. (2016) track attention history over source words to guide subsequent attention and alleviate under- and over-translation. Deliberation Networks (Xia et al., 2017) use a first decoder to generate an initial sequence and a second decoder to refine it using information from the complete draft. CueRefiner estimates semantic cue coverage after the initial decoding pass by comparing cue representations with decoder states. It combines this estimate with cue confidence to control bounded, temporally weighted residual updates to visual features, then reuses the same frozen decoder to generate the final translation. This connects estimated semantic coverage to refinement of the visual evidence supplied to the translator.

Our frozen-backbone design is also related to parameter-efficient adaptation. Adapters (Houlsby et al., 2019) learn task-specific modules while retaining pretrained parameters, and BLIP-2 (Li et al., 2023) trains a query-based interface between frozen visual and language models. RefineSLT uses trainable modules to ground semantic cues and selectively refine the temporal features supplied to an existing SLT decoder.

3 METHODOLOGY

3.1 OVERVIEW

End-to-end sign language translation (SLT) systems can produce fluent sentences while omitting or mistranslating a small number of concepts that determine the meaning of an utterance. Correcting this failure requires answering three questions: what content constitutes a *semantic cue*, where in the video its supporting evidence appears, and how that evidence can be strengthened without disrupting an otherwise correct translation. We introduce REFINESLT, a lightweight framework that addresses these questions with three components. During training, *CueMiner* converts each target sentence into a compact set of cue pseudo-labels by estimating the semantic effect of removing candidate

words or phrases. Given the temporal encoder features and an initial decoding pass, ***CueGrounder*** learns to recover these cues from the video and locate their temporal evidence. ***CueRefiner*** then estimates how well each grounded cue is represented in the initial translation and modulates feature updates using this estimate together with cue confidence. At inference, CueMiner is removed; CueGrounder and CueRefiner operate between the encoder and the original decoder using only the video and the backbone's own initial prediction.

3.2 Problem Formulation

Given a signing video V, a pretrained SLT backbone consists of a visual encoder E_ϕ and a text decoder D_ϕ. The encoder produces a temporal representation $H = [h_1, \ldots, h_T] \in \mathbb{R}^{T\times d}$. An initial decoding pass over H produces a translation y^0 and its decoder states $U^0 = [u_1^0, \ldots, u_N^0]$. CueGrounder and CueRefiner take (H, U^0) as input and predict a localized residual ΔH, yielding $\widetilde{H} = H + \Delta H$. The same decoder then generates the final translation $\widehat{y} = D_\phi(\widetilde{H})$. We keep the pretrained backbone frozen by default, allowing the proposed components to learn cue-sensitive refinement while preserving the backbone's general translation ability.

3.3 Semantic Cue Mining with CueMiner

In this work, we define a *semantic cue* as a context-dependent word or short phrase that conveys information central to a sentence and whose omission or mistranslation can materially alter the intended message. This definition is informed by frame-semantic accounts of words and phrases in context (Das et al., 2014) and studies of meaning-altering translation errors (Karpinska et al., 2022). This definition is instance-specific: corpus rarity alone does not imply importance, and a common expression may become critical in a particular context. CueMiner operationalizes this definition as a deterministic, training-only teacher. It requires no manual cue annotation and introduces no trainable parameters into the SLT model.

Given a target sentence y^*, CueMiner constructs a candidate set $\mathcal{C}(y^*)$ containing content-bearing words and contiguous phrases of up to L tokens. It removes punctuation, stopword-only spans, and duplicate surface forms. For each remaining candidate a, CueMiner masks the complete span and measures the resulting change in sentence meaning with a frozen sentence encoder $\mathcal{T}$, using a masking-based importance criterion related to MDERank (Zhang et al., 2022). This contextual deletion effect is combined with an IDF prior estimated exclusively from the training targets:

$$s(a; y^*) = \lambda_{\text{sem}}\ \mathrm{Norm}_{y^*}[1 - \cos(\mathcal{T}(y^*), \mathcal{T}(\mathrm{mask}(y^*, a)))] + \lambda_{\text{idf}}\ \mathrm{Norm}_{y^*}[\mathrm{IDF}(a)]\,. \quad (1)$$

Here, Norm_{y^*} denotes min–max normalization over candidates from the same sentence. For a multi-token phrase, $\mathrm{IDF}(a)$ is the mean IDF of its constituent content words, which avoids favoring a phrase merely because its exact surface form is rare. The deletion term is the primary measure of semantic importance, whereas IDF acts as a lightweight specificity prior when candidates have similar contextual effects.

CueMiner ranks the candidates by $s(a; y^*)$ and applies redundancy suppression before selecting the top K. A lower-ranked candidate is discarded when it substantially overlaps an already selected span or has a near-duplicate text embedding. This prevents several surface variants of the same concept from occupying multiple cue slots. The resulting set $A^* = \mathrm{SelectTopK}(\mathcal{C}(y^*), s)$ provides cue pseudo-labels for CueGrounder. All masked variants are encoded in batches and the cue sets are precomputed once, so CueMiner adds only offline preprocessing cost. It provides no temporal annotation and is never used at inference.

3.4 Semantic Cue Grounding with CueGrounder

CueGrounder converts sentence-level cue supervision into video-conditioned cue representations. It is placed after the visual encoder and additionally observes the decoder states from the initial translation. Given (H, U^0), it predicts a small set of cue slots $\{(q_i, p_i, \alpha_i)\}_{i=1}^K$, where q_i represents the cue semantics, p_i is its confidence, and α_i describes its temporal support. This representation explicitly separates what concept is important from where its visual evidence occurs.

Cue discovery. CueGrounder uses learnable cue queries to attend jointly to H and U^0. The visual pathway enables recovery of concepts that may be absent from y^0, while the decoder states provide

sentence context and indicate what the backbone has already expressed. Following the set-prediction formulation with bipartite matching used in DETR (Carion et al., 2020), we treat semantic cues as an unordered set and match the predicted slots to the CueMiner pseudo-labels A^* in a frozen text-embedding space. Matched slots are encouraged to approach their target cue embeddings and predict high confidence, whereas unmatched slots are trained as empty. This set-based supervision supports a variable number of cues without imposing an artificial order on their importance.

Temporal grounding. CueGrounder treats each predicted cue as a query over the temporal visual sequence:

$$\alpha_{i,t} = \mathrm{softmax}_t\left(\frac{(W_q q_i)^\top (W_h h_t)}{\sqrt{d_c}}\right), \qquad z_i = \sum_{t=1}^{T} \alpha_{i,t} h_t, \tag{2}$$

where z_i summarizes the localized visual evidence for cue i. During training, the pooled evidence is aligned with the matched cue embedding using a cross-modal contrastive objective, with nonmatching cues in the minibatch serving as negatives. This encourages α_i to emphasize visual content that distinguishes the target concept without treating the learned attention as an annotated temporal boundary. CueGrounder therefore outputs predicted semantic cue representations together with the localized visual evidence used for refinement.

3.5 Cue-Guided Refinement with CueRefiner

A detected cue should not automatically change the translation: it may already be expressed correctly, or its visual evidence may be uncertain. CueRefiner therefore first compares each cue representation q_i with the initial decoder states U^0. Their maximum semantic compatibility defines a coverage score ρ_i. The intervention gate g_i combines this coverage with the cue confidence, assigning a larger correction strength only to reliable cues that are insufficiently represented in the initial translation. CueRefiner then maps the cue, its localized evidence, and the initial sentence context to a residual vector and distributes it across temporal positions according to the grounding weights:

$$\begin{aligned} \rho_i &= \max_j \sigma\big((W_c q_i)^\top (W_u u_j^0)\big), \qquad g_i = p_i(1-\rho_i), \\ \widetilde{h}_t &= h_t + \eta \tanh\left(\sum_{i=1}^{K} g_i \alpha_{i,t} R_\theta(q_i, z_i, \bar{u}^0)\right), \end{aligned} \tag{3}$$

where $\bar{u}^0$ summarizes the initial sentence context and η bounds the update magnitude. The gate suppresses cues that are uncertain or already covered, while the temporal weights concentrate updates on the estimated supporting evidence. The residual connection retains the original features as the basis for refinement. CueRefiner therefore does not rewrite the sentence directly; it improves the representation from which the original decoder produces the final translation.

3.6 Training and Inference

Training. CueMiner first processes all training targets offline and produces a fixed cue set A^* for each sample. We then optimize the neural components in three stages to prevent unstable cue predictions from corrupting feature refinement. First, the SLT backbone is frozen and CueGrounder learns cue discovery and temporal grounding from the CueMiner pseudo-labels and paired video–text alignment. Second, CueGrounder is frozen and CueRefiner is trained through the fixed decoder, optimizing its localized residuals for translation quality while regularizing their magnitude and sparsity. Third, CueGrounder and CueRefiner are jointly fine-tuned with a smaller learning rate, while the backbone remains frozen. The overall objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{SLT}} + \lambda_c \mathcal{L}_{\mathrm{cue}} + \lambda_a \mathcal{L}_{\mathrm{align}} + \lambda_r \mathcal{L}_{\mathrm{reg}}, \tag{4}$$

where $\mathcal{L}_{\mathrm{SLT}}$ supervises the final translation, $\mathcal{L}_{\mathrm{cue}}$ trains set-level cue discovery from CueMiner pseudo-labels, $\mathcal{L}_{\mathrm{align}}$ associates each cue with localized video evidence, and $\mathcal{L}_{\mathrm{reg}}$ penalizes unnecessarily large or dense interventions. The target sentence determines the supervision through CueMiner but is never supplied as an input to CueGrounder or CueRefiner.

Inference. CueMiner and the target sentence are absent at inference. Given only V, the backbone first produces H, y^0, and U^0. CueGrounder predicts likely semantic cues together with their temporal evidence, CueRefiner selectively strengthens cues that are not adequately covered by y^0, and the unchanged decoder generates $\widehat{y}$ from $\widetilde{H}$. Low cue confidence or high estimated coverage suppresses the residual updates. When the updates are zero, the model reduces to the original backbone under the same decoding configuration. The framework can therefore be attached to any SLT architecture that exposes temporal encoder features and supports a second decoding pass, although CueGrounder and CueRefiner are trained for the selected backbone.

4 EXPERIMENTS

Experimental Setup. We evaluate **RefineSLT** across multiple sign language datasets and architectures to assess the effectiveness and generality of cue-guided semantic refinement. Experiments are conducted on three benchmarks: **PHOENIX-2014T** (Camgoz et al., 2018) (German Sign Language), **How2Sign** (Duarte et al., 2021) (American Sign Language), and **Auslan-Daily** (Shen et al., 2023) (Australian Sign Language), including its Communication and News subsets. To focus the evaluation on longer utterances, we exclude test samples whose reference translations contain three or fewer words and use the same filtered test subsets for all compared methods. Unless otherwise stated, all subsequent test results refer to these filtered subsets, which we identify by their original dataset names. We evaluate seven representative SLT backbones: *MMTLB* (Chen et al., 2022a), *GASLT* (Yin et al., 2023), *TwoStreamNetwork* (Chen et al., 2022b), *CV-SLT* (Zhao et al., 2024), *Sign2GPT* (Wong et al., 2024), *LiTFiC* (Jang et al., 2025), and *MMSLT* (Kim et al., 2025). For each backbone X, we train CueGrounder and CueRefiner using offline supervision from CueMiner while keeping the backbone encoder and decoder frozen. Evaluation follows standard metrics, reporting **ROUGE-L** (Lin, 2004) and **BLEU-1–4** (Papineni et al., 2002), consistent with prior SLT studies.

Implementation Details. All models follow official data splits and identical preprocessing, tokenization, and evaluation settings for fairness. Optimization uses AdamW (Loshchilov, 2017) with learning rate 2×10^{-5}, batch size 32, and weight decay 0.01 for 30 epochs, applying early stopping by BLEU-4 on the validation set. Inference consists of an initial decoding pass, cue-guided feature refinement, and a second pass through the same decoder. Experiments are run on one NVIDIA A100 GPU under PyTorch 2.2 with mixed-precision training. Video frames (25 fps) are encoded by a pre-trained I3D (Carreira & Zisserman, 2017) backbone, while pose features are extracted using *HRNet* (Wang et al., 2021) trained on COCO-WholeBody (Jin et al., 2020).

4.1 COMPARISON EXPERIMENT

Tables 1 and 2 compare RefineSLT with its underlying SLT backbones on the evaluation sets. For each backbone, we train CueGrounder and CueRefiner while keeping the original encoder and decoder frozen; CueMiner provides offline training supervision and is removed at inference. On PHOENIX-2014T, RefineSLT improves test BLEU-4 and ROUGE-L across all seven backbones. Specifically, BLEU-4 increases from 15.62 to 17.70 for GASLT and from 28.15 to 31.56 for CV-SLT, corresponding to relative improvements of 13.3% and 12.1%, respectively. These gains demonstrate effectiveness across different levels of baseline translation quality. On How2Sign, all seven backbones improve across the five test metrics, further supporting the applicability of cue-guided refinement. On Auslan-Daily Communication, BLEU-4 increases from 7.16 to 11.04 for CV-SLT and from 8.96 to 12.67 for Sign2GPT, yielding relative improvements of 54.2% and 41.4%, respectively. All six backbones also improve across the five metrics on the News subset. Together, these results show that grounding semantic cues and using them to guide visual feature refinement can enhance existing translators across the evaluated architectures, sign languages, and domains, with refinement modules trained separately for each backbone.

4.2 ABLATION STUDY

We evaluate RefineSLT with TwoStreamNetwork on the PHOENIX-2014T test set under identical backbone, data, decoding, and optimization settings, using $K = 2$ unless specified otherwise. Reference translations and CueMiner outputs are used only for training supervision. Table 3 shows that the complete model improves BLEU-4 from 27.62 to 29.88 and ROUGE-L from 53.08 to 56.92, corresponding to relative gains of 8.2% and 7.2%. A parameter-matched generic adapter without

	PHOENIX-2014T										How2Sign (ASL)									
	Dev					Test					Dev					Test				
	R	B1	B2	B3	B4	R	B1	B2	B3	B4	R	B1	B2	B3	B4	R	B1	B2	B3	B4
MMTLB(RGB)	51.49	51.76	39.01	30.98	25.44	52.44	52.73	40.11	32.23	26.87	25.17	23.54	15.12	10.36	8.47	23.27	21.10	14.13	9.43	7.98
RefineSLT [MMTLB]	**54.18**	**55.62**	**42.54**	**33.36**	**28.73**	**55.16**	**56.58**	**43.76**	**35.76**	**28.31**	**27.42**	**26.21**	**17.46**	**11.89**	**9.57**	**25.98**	**24.18**	**16.51**	**10.62**	**8.92**
GASLT(RGB)	40.67	39.37	26.35	19.54	15.42	39.63	38.77	26.50	19.71	15.62	15.08	16.70	11.89	7.47	3.95	13.69	14.10	10.25	5.46	2.83
RefineSLT [GASLT]	**43.37**	**44.86**	**30.40**	**22.80**	**18.34**	**42.40**	**44.22**	**29.74**	**22.16**	**17.70**	**18.86**	**19.22**	**13.44**	**9.13**	**5.21**	**16.33**	**16.81**	**12.14**	**6.83**	**4.55**
TwoStreamNetwork	53.27	53.38	40.66	32.66	27.21	53.08	53.46	41.09	33.11	27.62	33.74	32.96	20.24	12.08	9.06	31.88	31.47	19.02	11.08	8.23
RefineSLT [TSN]	**57.75**	**57.94**	**45.04**	**35.94**	**30.42**	**56.92**	**57.02**	**45.52**	**35.44**	**29.88**	**36.81**	**36.68**	**23.31**	**14.65**	**11.17**	**35.12**	**35.71**	**21.63**	**12.14**	**10.11**
CV-SLT	53.89	55.30	42.30	34.07	28.38	52.83	54.27	41.88	33.77	28.15	32.41	32.91	20.16	13.55	10.82	30.73	31.68	19.14	12.49	7.91
RefineSLT [CV-SLT]	**58.51**	**60.10**	**46.84**	**38.48**	**32.72**	**56.51**	**58.05**	**45.51**	**37.27**	**31.56**	**36.73**	**36.17**	**23.14**	**15.25**	**11.98**	**33.94**	**34.72**	**23.52**	**13.85**	**10.02**
Sign2GPT	42.95	37.51	25.55	21.17	16.31	41.86	35.73	24.93	18.73	14.13	27.19	28.95	15.40	8.63	6.29	25.53	28.81	14.92	8.83	5.75
RefineSLT [Sign2GPT]	**46.29**	**42.32**	**30.22**	**25.08**	**20.95**	**45.28**	**40.01**	**27.30**	**22.42**	**18.52**	**33.14**	**34.22**	**19.61**	**12.92**	**9.82**	**31.88**	**32.97**	**18.91**	**11.34**	**8.16**
LiTFiC	48.36	49.59	37.12	31.52	26.15	46.36	47.10	35.48	27.79	21.73	31.26	33.43	19.61	15.02	11.89	29.99	34.26	18.47	13.74	10.62
RefineSLT [LiTFiC]	**52.88**	**54.91**	**42.54**	**36.08**	**30.57**	**50.41**	**52.06**	**39.96**	**30.91**	**25.28**	**35.47**	**37.02**	**23.07**	**18.19**	**13.74**	**34.74**	**35.84**	**22.71**	**16.99**	**11.63**
MMSLT	44.83	43.93	34.02	27.43	22.78	44.03	44.22	34.47	27.81	23.18	34.49	38.03	24.45	16.74	10.67	31.06	37.15	21.22	13.83	9.00
RefineSLT [MMSLT]	**48.02**	**48.45**	**38.01**	**31.26**	**25.49**	**47.11**	**47.51**	**38.36**	**30.74**	**24.66**	**38.22**	**42.06**	**28.18**	**19.92**	**13.21**	**36.44**	**40.21**	**23.88**	**16.62**	**11.42**

Table 1: Comparison with state-of-the-art methods without gloss supervision on **PHOENIX-2014T** and **How2Sign**. 'R' denotes ROUGE-L; 'B1–B4' denote BLEU-1 to BLEU-4.

	Auslan-Daily Communication					Auslan-Daily News				
	R	B1	B2	B3	B4	R	B1	B2	B3	B4
MMTLB(RGB)	17.64	18.35	14.27	9.76	6.11	18.90	19.64	5.30	3.26	2.31
RefineSLT [MMTLB]	**25.98**	**27.43**	**17.02**	**12.35**	**7.39**	**20.52**	**21.04**	**10.11**	**7.38**	**4.82**
GASLT(RGB)	29.18	22.61	8.53	4.91	3.50	22.01	19.54	7.45	4.41	2.56
RefineSLT [GASLT]	**33.93**	**29.25**	**12.87**	**9.08**	**7.60**	**28.64**	**27.88**	**11.71**	**8.02**	**6.13**
TwoStreamNetwork	34.66	32.61	19.23	10.53	6.07	26.74	27.20	10.02	7.42	3.56
RefineSLT [TSN]	**38.32**	**39.35**	**24.47**	**14.94**	**9.04**	**33.12**	**34.18**	**15.58**	**11.91**	**7.02**
CV-SLT	32.97	32.67	19.40	11.78	7.16	30.86	35.73	12.03	9.40	4.54
RefineSLT [CV-SLT]	**36.43**	**37.85**	**26.87**	**16.94**	**11.04**	**33.58**	**37.40**	**13.68**	**10.73**	**4.92**
Sign2GPT	33.01	32.11	17.02	11.67	8.96	23.19	20.36	8.06	5.22	3.65
RefineSLT [Sign2GPT]	**39.18**	**38.35**	**21.74**	**15.50**	**12.67**	**30.42**	**27.88**	**12.96**	**9.37**	**7.28**
MMSLT	35.26	35.83	20.59	13.96	11.26	24.52	23.45	11.23	9.19	5.84
RefineSLT [MMSLT]	**40.41**	**41.70**	**25.36**	**18.65**	**14.48**	**30.18**	**29.27**	**16.82**	**13.31**	**9.42**

Table 2: Comparison between backbone models and our method on the Communication and News subsets of Auslan-Daily

cue supervision obtains only 27.10 BLEU-4, indicating that additional trainable capacity alone does not explain these gains.

Replacing CueMiner with random content-span or IDF-only selection yields 28.13 and 28.44 BLEU-4, respectively. Both outperform the backbone, but the complete CueMiner performs best across all five metrics, supporting supervision based on contextual semantic importance. Replacing cue-specific temporal grounding with global visual pooling reduces BLEU-4 to 23.30 and ROUGE-L to 48.49, below the unchanged backbone. Removing the missing-cue check similarly lowers BLEU-4 to 27.39 and correction success from 92% to 76%. Together, these results indicate that effective refinement requires both associating cues with temporal visual evidence and accounting for content already expressed in the initial translation.

The cue budget further affects translation quality and correction reliability. With $K = 1$, RefineSLT achieves 28.03 BLEU-4 and the highest correction success rate of 93%. Increasing K to 2 raises BLEU-4 to 29.88 while retaining 92% success, whereas $K = 3$ reduces these scores to 29.03 and 88%. Thus, $K = 2$ achieves the best performance across all five translation metrics among the evaluated budgets, balancing content recovery with correction reliability.

4.3 Qualitative Analysis

Figure 3 illustrates how RefineSLT recovers meaning-critical content across backbones and sign languages. On PHOENIX-2014T, it corrects July 4 to July 12 and recovers the omitted phrase "brief thunderstorms." On Auslan-Daily, it restores "Sally" and "paint" in the Communication subset, and corrects "sunny" to "cloudy" while recovering "cold" in News. These changes improve specific semantic facts, consistent with CueGrounder's role in associating predicted cues with visual evidence and CueRefiner's use of estimated coverage to guide feature updates. Recovery remains partial in some cases: the ABC example restores introductory content but retains "thank you for your com-

Table 3: Ablation study of REFINESLT with TwoStreamNetwork on the PHOENIX-2014T test set. • and ○ indicate enabled and disabled design choices, respectively. Success is the percentage of attempted cue corrections for which the final translation contains the corresponding reference cue under normalized exact matching.

Generic Adapter	K	CueMiner	CueGrounder	CueRefiner		Translation Quality					Success ↑
				Missing Check	Correction	R	B1	B2	B3	B4	
○	–	–	○	○	○	53.08	53.46	41.09	33.11	27.62	–
•	–	–	○	○	○	52.76	53.21	41.37	33.58	27.10	–
○	2	R	•	•	•	53.61	54.06	41.40	33.52	28.13	84%
○	2	IDF	•	•	•	55.37	55.49	42.81	34.36	28.44	86%
○	2	•	○	•	•	48.49	49.54	38.71	29.16	23.30	81%
○	2	•	•	○	•	54.68	55.18	41.99	33.56	27.39	76%
○	1	•	•	•	•	54.96	55.70	43.00	34.18	28.03	**93%**
○	2	•	•	•	•	**56.92**	**57.02**	**45.52**	**35.44**	**29.88**	92%
○	3	•	•	•	•	55.37	56.85	44.13	34.52	29.03	88%

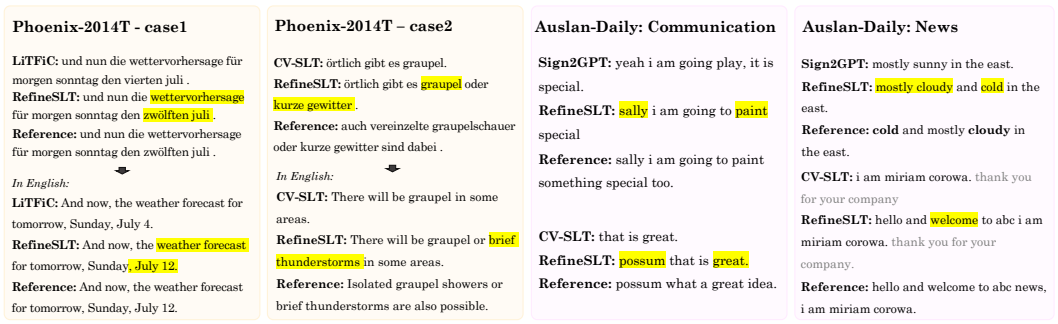


Figure 3: Qualitative comparisons on PHOENIX-2014T and the Communication and News subsets of Auslan-Daily. Each example shows a backbone output, its RefineSLT counterpart, and the reference. Highlighted spans indicate selected semantic content in the refined outputs; English translations accompany the German examples.

pany," which is absent from the reference. These examples illustrate both the value of cue-guided refinement and the limits of its selectivity.

5 CONCLUSION

We introduced RefineSLT, a framework for enhancing pretrained sign language translators through semantic cues. CueMiner provides text-derived supervision, CueGrounder predicts cues and their temporal evidence, and CueRefiner controls feature updates using cue confidence and estimated coverage in the initial translation. The backbone remains frozen and its original decoder generates the final translation from the refined features. Together, these components connect what semantic content to recover, where to seek visual evidence, and when to refine, providing a unified formulation for selective semantic enhancement.

6 DISCUSSION

RefineSLT depends on the semantic information retained by the frozen backbone and the reliability of learned cue alignment. Its refinement gate relies on estimated cue confidence and translation coverage, both of which may be inaccurate, particularly under domain shift. These estimates do not guarantee that a feature update improves the translation. Future work should examine richer semantic relations, robustness across signing domains, and evaluation by sign-language users to assess both recovered meaning and harmful corrections.

Reproducibility Statement

To support reproducibility, we provide the source code for RefineSLT in the ZIP archive submitted as supplementary material. The code will also be made publicly available on GitHub following publication of the paper.

References

Kamil Bennani-Smires, Claudiu Musat, Andreea Hossmann, Michael Baeriswyl, and Martin Jaggi. Simple unsupervised keyphrase extraction using sentence embeddings. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pp. 221–229. Association for Computational Linguistics, 2018. doi: 10.18653/v1/K18-1022. URL https://aclanthology.org/K18-1022/.

Hannah Bull, Triantafyllos Afouras, Gül Varol, Samuel Albanie, Liliane Momeni, and Andrew Zisserman. Aligning subtitles in sign language videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11552–11561, 2021.

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

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 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*, pp. 10023–10033, 2020.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020*, pp. 213–229. Springer, 2020. doi: 10.1007/978-3-030-58452-8_13. URL https://doi.org/10.1007/978-3-030-58452-8_13.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Carreira_Quo_Vadis_Action_CVPR_2017_paper.html.

Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen Lin. A simple multi-modality transfer learning baseline for sign language translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5120–5130, 2022a.

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

Yiting Cheng, Fangyun Wei, Jianmin Bao, Dong Chen, and Wenqiang Zhang. CiCo: Domain-aware sign language retrieval via cross-lingual contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19016–19026, 2023.

Dipanjan Das, Desai Chen, André FT Martins, Nathan Schneider, and Noah A Smith. Frame-semantic parsing. *Computational linguistics*, 40(1):9–56, 2014.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. How2Sign: A large-scale multimodal dataset for continuous american sign language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2735–2744, 2021.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. Evaluating models' local decision boundaries via contrast sets. In

Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 1307–1323. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.findings-emnlp.117. URL https://aclanthology.org/2020.findings-emnlp.117/.

Jia Gong, Lin Geng Foo, Yixuan He, Hossein Rahmani, and Jun Liu. Llms are good sign language translators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18362–18372, 2024.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

Youngjoon Jang, Haran Raajesh, Liliane Momeni, Gül Varol, and Andrew Zisserman. Lost in translation, found in context: Sign language translation with contextual cues. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8742–8752, 2025.

Peiqi Jiao, Yuecong Min, and Xilin Chen. Visual alignment pre-training for sign language translation. In *European Conference on Computer Vision*, 2024.

Sheng Jin, Lumin Xu, Jin Xu, Can Wang, Wentao Liu, Chen Qian, Wanli Ouyang, and Ping Luo. Whole-body human pose estimation in the wild. In *European Conference on Computer Vision*, pp. 196–214. Springer, 2020.

Marzena Karpinska, Nishant Raj, Katherine Thai, Yixiao Song, Ankita Gupta, and Mohit Iyyer. Demetr: Diagnosing evaluation metrics for translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 9540–9561, 2022.

Jungeun Kim, Hyeongwoo Jeon, Jongseong Bae, and Ha Young Kim. Leveraging the power of MLLMs for gloss-free sign language translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 21048–21058, 2025.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81. Association for Computational Linguistics, 2004. URL https://aclanthology.org/W04-1013/.

Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. 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)*, pp. 12904–12916. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.722. URL https://aclanthology.org/2023.acl-long.722/.

Yuqi Liu, Wenqian Zhang, Sihan Ren, Chengyu Huang, Jingyi Yu, and Lan Xu. Scope: Sign language contextual processing with embedding from llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5739–5747, 2025.

I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Yuecong Min, Aiming Hao, Xiujuan Chai, and Xilin Chen. Visual alignment constraint for continuous sign language recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11542–11551, 2021.

Liliane Momeni, Gül Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. Watch, read and lookup: Learning to spot signs from multiple supervisors. In *Asian Conference on Computer Vision*, 2020.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318. Association for Computational Linguistics, 2002. URL https://aclanthology.org/P02-1040/.

Junfu Pu, Wengang Zhou, and Houqiang Li. Iterative alignment network for continuous sign language recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4165–4174, 2019.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4902–4912. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.442. URL https://aclanthology.org/2020.acl-main.442/.

Xin Shen, Shaozu Yuan, Hongwei Sheng, Heming Du, and Xin Yu. Auslan-daily: Australian sign language translation for daily communication and news. *Advances in Neural Information Processing Systems*, 36:80455–80469, 2023.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 76–85. Association for Computational Linguistics, 2016. doi: 10.18653/v1/P16-1008. URL https://aclanthology.org/P16-1008/.

Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2021.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. Sign2GPT: Leveraging large language models for gloss-free sign language translation. In *International Conference on Learning Representations*, 2024.

Yingce Xia, Fei Tian, Lijun Wu, Jianxin Lin, Tao Qin, Nenghai Yu, and Tie-Yan Liu. Deliberation networks: Sequence generation beyond one-pass decoding. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/c6036a69be21cb660499b75718a3ef24-Abstract.html.

Aoxiong Yin, Tianyun Zhong, Li Tang, Weike Jin, Tao Jin, and Zhou Zhao. Gloss attention for gloss-free sign language translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2551–2562, 2023.

Linhan Zhang, Qian Chen, Wen Wang, Chong Deng, ShiLiang Zhang, Bing Li, Wei Wang, and Xin Cao. MDERank: A masked document embedding rank approach for unsupervised keyphrase extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 396–409. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.findings-acl.34. URL https://aclanthology.org/2022.findings-acl.34/.

Yuan Zhang, Jason Baldridge, and Luheng He. PAWS: Paraphrase adversaries from word scrambling. 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)*, pp. 1298–1308. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1131. URL https://aclanthology.org/N19-1131/.

Rui Zhao, Liang Zhang, Biao Fu, Cong Hu, Jinsong Su, and Yidong Chen. Conditional variational autoencoder for sign language translation with cross-modal alignment. In *Proceedings of the aaai conference on artificial intelligence*, volume 38, pp. 19643–19651, 2024.

Jiangbin Zheng, Yidong Chen, Chong Wu, Xiaodong Shi, and Suhail Muhammad Kamal. Enhancing neural sign language translation by highlighting the facial expression information. *Neurocomputing*, 464:462–472, 2021.

Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1316–1325, 2021.

Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. Spatial-temporal multi-cue network for sign language recognition and translation. *IEEE transactions on multimedia*, 24:768–779, 2022.