SIGNRAG: UNIFIED RETRIEVAL-AUGMENTED GLOSS-FREE SIGN LANGUAGE TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Contemporary decoder-only large language models (LLMs) have demonstrated strong capabilities across a wide range of domains. However, existing pretraining paradigms for gloss-free sign language translation (SLT) are largely designed around conventional encoder-decoder pretrained language models, which limits their direct applicability to decoder-only LLMs. To address this limitation, we propose SignRAG, a unified framework combining hierarchical pretraining, target-domain retrieval augmentation, and retrieval-aware reinforcement fine-tuning. Hierarchical pretraining first learns linguistically grounded sign representations and then jointly aligns the sign encoder with an LLM, mitigating cross-modal optimization imbalance. For downstream adaptation, SignRAG complements parameter-based fine-tuning with a target-domain retrieval gallery that provides instance-specific translation cues. To ensure that retrieved contexts are used appropriately, we further introduce Retrieval Utility-Guided Reinforcement Fine-Tuning (RUG-RFT), which combines translation-quality and retrieval-utility rewards to encourage beneficial retrieval use while suppressing harmful reliance. Experiments on multiple SLT benchmarks establish new state-of-the-art performance. In particular, to the best of our knowledge, SignRAG is the first gloss-free approach to outperform gloss-supervised methods across all reported metrics on CSL-Daily. Our code and models of different sizes will be publicly released.

1 INTRODUCTION

Sign language translation (SLT) aims to translate visual sign sequences into natural language. Despite recent progress, existing methods still struggle to generalize across datasets with different signers, signing styles, and vocabulary distributions. A natural solution is to pretrain on large-scale sign language corpora and subsequently adapt the model to downstream SLT benchmarks.

Existing pretraining paradigms for gloss-free SLT can be broadly divided into two categories. *Encoder-only pretraining* independently pretrains the sign encoder before transferring it to downstream translation models (Zhou et al., 2023; Chen et al., 2024; Wong et al., 2024; Chen et al., 2025; Jiao et al., 2024), whereas *end-to-end pretraining* jointly optimizes visual and language components on large-scale sign-language corpora (Uthus et al., 2023; Zhang et al., 2024; Tanzer & Zhang, 2025; Li et al., 2025b; Fish & Bowden, 2026). However, most existing methods still rely on T5-based encoder–decoder models, whose language modeling capacity remains limited compared with contemporary LLMs.

Replacing these decoders with a substantially stronger LLM is not a drop-in upgrade. Because the language branch is already strongly pretrained, it can dominate optimization and leave the comparatively weaker visual pathway under-trained. We refer to this phenomenon as **Cross-Modal Optimization Imbalance**. Such imbalance has been widely observed in multimodal learning (Wang et al., 2020; Huang et al., 2022; Peng et al., 2022) and can further induce shortcut learning driven by strong language priors (Goyal et al., 2017). It raises a key question: *how should large-scale sign pretraining be organized so that visual pathway still learns semantically grounded representations?*

We therefore propose a **Hierarchical Pretraining** strategy for LLM-based SLT. Inspired by the staged-alignment paradigm of general-purpose multimodal large language models (MLLMs) (Liu et al., 2023; Ye et al., 2024; Bai et al., 2025; McKinzie et al., 2024), we organize large-scale pretraining into two stages. We first use fine-grained sign–text supervision to learn linguistically grounded

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.

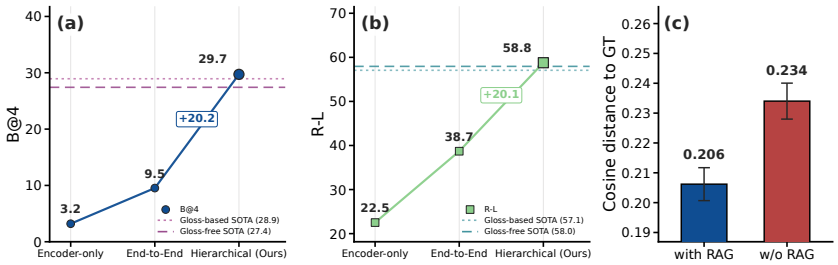


Figure 1: **Motivation.** Hierarchical pretraining and retrieval augmentation improve sign-to-language learning. **(a)**, **(b)** Comparison of different pretraining strategies. The solid and dashed lines denote the previous state-of-the-art performance under the gloss-free and gloss-based settings, respectively, corresponding to GeoSign (Fish & Bowden, 2026) and CV-SLT (Zhao et al., 2024). **(c)** Comparison of the semantic distance between generated and ground-truth (GT) texts with and without RAG. Semantic distance is measured using the independent sentence encoder *paraphrase-multilingual-MiniLM-L12-v2*. Bars report the mean cosine distance over all samples, with error bars denoting $\pm$SEM. Samples are drawn from the CSL-Daily dev set.

sign representations, and then jointly pretrain the sign encoder, multimodal projector, and LLM on the same corpus. In this way, the model first learns *what the sign means* and then learns *how to express its semantics in natural language*. As shown in Fig. 1(a) and (b), hierarchical pretraining clearly outperforms alternative pretraining schemes, while the `Pretrain`$\rightarrow$`SFT` variant alone already surpasses previous state-of-the-art methods under both gloss-free and gloss-based settings.

After large-scale pretraining, downstream adaptation introduces a second challenge, which we term the **Target-Domain Adaptation Dilemma**. Sign-language datasets exhibit substantial distribution shifts across languages, recording conditions, and signers, while downstream training sets are typically much smaller than the pretraining corpus. Aggressive fine-tuning may therefore erode transferable knowledge acquired during pretraining, whereas conservative fine-tuning may leave the target domain insufficiently adapted. To address this dilemma, we complement parametric SFT with a **non-parametric target-domain semantic memory**. Specifically, we construct a retrieval gallery exclusively from the target-domain training split and retrieve semantically related sign instances together with their translations as instance-specific linguistic cues. As shown in Fig. 1(c), retrieval augmentation reduces the semantic distance between generated and GT texts from 0.234 to 0.206.

Retrieval, however, is not always beneficial, leading to a third challenge that we term the **Retrieval Reliance Dilemma**. Standard SFT optimizes token-level likelihood but provides no explicit supervision on whether retrieved contexts actually benefit generation, making both underutilization and harmful reliance difficult to regulate. We first introduce retrieval dropout during retrieval-augmented SFT to improve robustness when retrieved information is absent. We then propose **Retrieval Utility-Guided Reinforcement Fine-Tuning (RUG-RFT)** based on Group Relative Policy Optimization (GRPO) (Shao et al., 2024). RUG-RFT combines translation-quality and retrieval-utility rewards, encouraging the model to exploit informative retrieval while suppressing harmful reliance on misleading contexts.

We conduct extensive experiments on four downstream SLT benchmarks: CSL-Daily, PHOENIX-2014T, How2Sign, and OpenASL. SignRAG consistently outperforms existing gloss-free state-of-the-art methods. On CSL-Daily, it further surpasses gloss-supervised methods across all metrics; to the best of our knowledge, this is the first gloss-free SLT approach to achieve such performance. Our main contributions are summarized as follows:

- We propose a hierarchical pretraining paradigm that first learns linguistically grounded sign representations through fine-grained sign–text alignment and then performs large-scale joint multimodal pretraining with a contemporary LLM.
- We introduce a target-domain retrieval-augmented adaptation mechanism that constructs a non-parametric semantic memory to complement parameter-based SFT with instance-specific target-domain knowledge.
- We propose Retrieval Utility-Guided Reinforcement Fine-Tuning (RUG-RFT), which jointly optimizes translation quality and retrieval contribution to enable more effective and robust use of retrieved contexts.
- Extensive experiments across multiple SLT benchmarks validate the effectiveness of the proposed framework and establish new state-of-the-art performance.

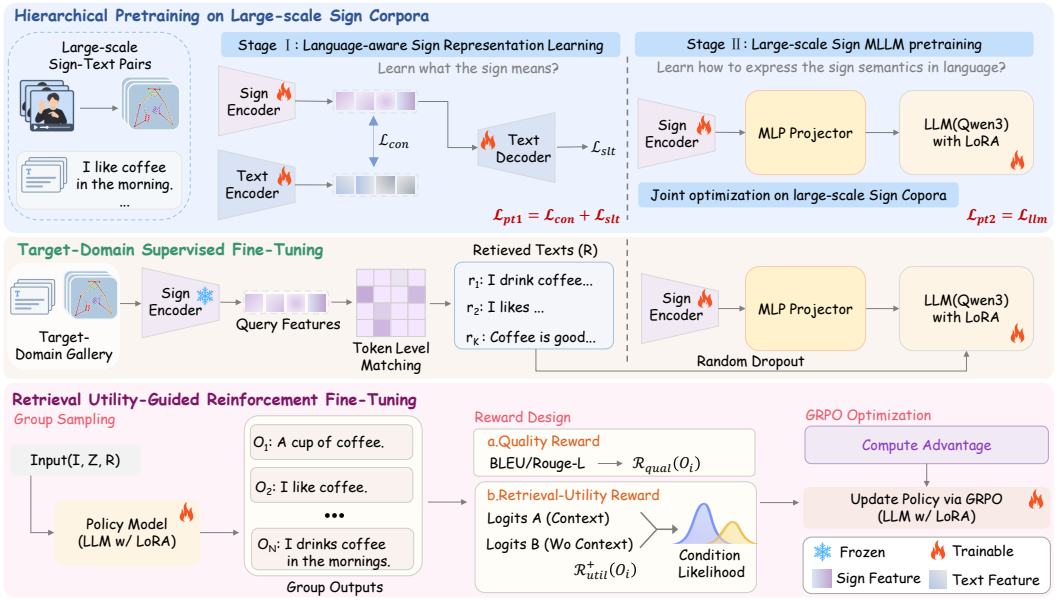


Figure 2: **Overview of SignRAG. (1)** Hierarchical pretraining on large-scale sign corpora: Stage I learns language-aware sign representations with a contrastive objective $\mathcal{L}_{\text{con}}$ and a skeleton-to-text objective $\mathcal{L}_{\text{slt}}$; Stage II couples the pretrained sign encoder with the LLM (Qwen3) via a multimodal projector and performs large-scale joint pretraining. **(2)** Target-domain supervised fine-tuning: a *sign retrieval gallery* provides semantically related reference translations as auxiliary cues, and the model is fine-tuned with sign features and retrieved texts, with random retrieval dropout. **(3)** Retrieval utility-guided reinforcement fine-tuning: a quality reward and a retrieval-utility reward are combined and optimized with GRPO.

2 Related Work

Sign language translation. Early SLT relies on gloss supervision via gloss recognition or cascaded translation (Camgoz et al., 2020; Zhou et al., 2021; Chen et al., 2022c;b), whereas gloss-free SLT maps sign videos directly to text using vision–language learning, LLM decoding, or cross-modal alignment (Camgoz et al., 2018; Li et al., 2020; Zhao et al., 2021; Yin et al., 2023; Zhou et al., 2023; Wong et al., 2024; Chen et al., 2024; Gong et al., 2024; Jiao et al., 2024; Chen et al., 2025), and external large-scale pretraining further improves transferability (Uthus et al., 2023; Rust et al., 2024; Li et al., 2025b). **Retrieval-augmented generation and reinforcement learning.** RAG conditions generation on retrieved non-parametric contexts (Lewis et al., 2020; Chen et al., 2022a), retrieval-based SLT (Yao et al., 2026) mainly targets gloss-based settings, and GRPO (Shao et al., 2024; Guo et al., 2025) has been applied to retrieval-enhanced reasoning and generation (Huang et al., 2025; Li et al., 2025a; Wang et al., 2025). SignRAG studies *how* large-scale sign pretraining should be organized under a modern LLM decoder, and complements parametric adaptation with a sign-to-text retrieval gallery and retrieval Utility-Guided reinforcement fine-tuning, without gloss annotations. A more detailed discussion is provided in Appendix A.

3 Method

As illustrated in Fig. 2, **SignRAG** consists of three stages for gloss-free sign language translation (SLT). **Stage 1** performs hierarchical pretraining, where Stage I learns linguistically grounded sign representations and Stage II jointly pretrains the sign encoder, multimodal projector, and LLM. **Stage 2** performs retrieval-augmented SFT using a target-domain *Sign Retrieval Gallery*, which retrieves semantically related sign instances and provides their paired translations as auxiliary cues; retrieval dropout is further applied during training. **Stage 3** applies **Retrieval Utility-Guided Reinforcement Fine-Tuning (RUG-RFT)** with GRPO, combining a Quality Reward and a Retrieval-Utility Reward to improve translation quality and encourage appropriate use of retrieved contexts. We next describe these three components in detail.

3.1 HIERARCHICAL PRETRAINING

Stage I: Sign–text alignment. We first align skeletal sign representations with natural-language semantics. Given a sign video V, MMPose (MMPose Contributors, 2020) first extracts 2D keypoints, which are subsequently encoded by a CoSign-based skeleton encoder (Jiao et al., 2023) into frame-level skeletal representations. The paired translation T is encoded by an mBART-based text encoder (Liu et al., 2020). The two encoders are jointly optimized with a bidirectional InfoNCE contrastive objective $\mathcal{L}_{\text{con}}$ (Cheng et al., 2023) and a CoCa-style skeleton-to-text translation objective $\mathcal{L}_{\text{slt}}$ (Yu et al., 2022):

$$\mathcal{L}_{\text{pt}} = \mathcal{L}_{\text{con}} + \mathcal{L}_{\text{slt}}. \tag{1}$$

Detailed formulations are provided in Appendix B.1. This stage endows the sign encoder with linguistically grounded representations before it is coupled with the LLM.

Stage II: Large-scale sign MLLM pretraining. We then couple the aligned sign encoder with an LLM. A GELU-activated MLP projector maps the skeletal features into the embedding space of Qwen3 (Yang et al., 2025), and LoRA adapters are attached to the query and value projections of the LLM self-attention layers. The sign encoder, the projector, and the LoRA-adapted LLM are jointly optimized on large-scale sign–text corpora, CSL-News (Li et al., 2025b) for Chinese sign language and YouTube-ASL (Uthus et al., 2023) for American sign language, with the autoregressive objective

$$\mathcal{L}_{\text{lm}} = -\sum_{t=1}^{T} \log p_\theta(y_t \mid y_{<t}, \mathbf{I}, \mathbf{Z}), \tag{2}$$

where $\mathbf{Z}$ denotes the projected sign features and $\mathbf{I}$ the instruction. Because the sign encoder enters joint training already grounded in language semantics, it no longer acts as the weak branch when coupled with the LLM, which mitigates the Cross-Modal Optimization Imbalance discussed in the Introduction.

3.2 RETRIEVAL-AUGMENTED SFT

Sign retrieval gallery. For each downstream dataset, we fine-tune the sign encoder on its training set and build a *sign-to-text* retrieval gallery from that split only; validation and test samples serve exclusively as queries, so that retrieved knowledge always comes from the target domain. For a gallery video V^g, its frame-level representation is

$$Z^g = \texttt{SkeletonEnc}(V^g) \in \mathbb{R}^{L_g \times C}, \tag{3}$$

where L_g denotes the temporal length and C is the feature dimension. Each gallery entry stores the encoded sign feature sequence and its paired ground-truth translation,

$$\mathcal{G}_n = (Z_n^g, T_n^g). \tag{4}$$

Query videos are encoded by the same skeleton encoder for sign-level retrieval.

Attention-weighted sign–sign similarity. Given a query sign sequence $Z^q \in \mathbb{R}^{L_q \times C}$ and a gallery sequence $Z^g \in \mathbb{R}^{L_g \times C}$, we first project their frame-level features into a shared D-dimensional embedding space:

$$Q = \text{Norm}(\phi_q(Z^q)), \qquad G = \text{Norm}(\phi_g(Z^g)), \tag{5}$$

where $Q \in \mathbb{R}^{L_q \times D}$ and $G \in \mathbb{R}^{L_g \times D}$. Here, $\phi_q(\cdot)$ and $\phi_g(\cdot)$ denote the projection heads, and $\text{Norm}(\cdot)$ performs row-wise ℓ_2 normalization.

We then compute the frame-level cosine similarity between the query and gallery sequences:

$$E_{i,j} = Q_i^\top G_j, \tag{6}$$

where $E \in \mathbb{R}^{L_q \times L_g}$. For each query frame, we normalize its similarities over the gallery sequence to obtain attention weights:

$$P_{i,j} = \frac{\exp(E_{i,j})}{\sum_{j'=1}^{L_g} \exp(E_{i,j'})}. \tag{7}$$

The global sign–sign similarity is obtained by aggregating the attention-weighted frame-level similarities:

$$S(Q, G) = \frac{1}{L_q} \sum_{i=1}^{L_q} \sum_{j=1}^{L_g} P_{i,j} E_{i,j}. \tag{8}$$

Compared with pooling an entire sign video into a single global representation, our formulation preserves fine-grained frame-level correspondences between sign sequences, which are crucial for retrieving semantically relevant samples. For clarity, the padding masks used to handle variable-length sequences are omitted from the above formulations. To improve memory efficiency, we shard the gallery across distributed workers and compute sign–sign similarities in query–gallery chunks. The exact global top-K results are then obtained by merging the local top-K candidates returned by all workers. Further implementation details are provided in Appendix B.4.

Retrieval-augmented fine-tuning. We use skeleton embeddings Z_s as the visual representation of sign language and project them into the LLM latent space:

$$Z = \mathrm{MLP}(Z_s), \tag{9}$$

where the projector is a GELU-activated MLP. We use Qwen3 (Yang et al., 2025) as the base LLM and form its input by concatenating the instruction $\mathbf{I}$, sign features $\mathbf{Z}$, and retrieved contexts $\mathbf{R}$.

To support the subsequent Retrieval-Utility Reward, we apply retrieval dropout during SFT:

$$\tilde{\mathbf{R}} = \begin{cases} \mathbf{R}, & \text{with probability } 1 - p, \\ \varnothing, & \text{with probability } p, \end{cases} \tag{10}$$

where $\varnothing$ denotes an empty token sequence without placeholder tokens. The model is fine-tuned with LoRA by minimizing

$$\mathcal{L}_{\mathrm{sft}} = -\sum_{t=1}^{T} \log p_\theta \Big(y_t \mid y_{<t}, \mathbf{I}, \mathbf{Z}, \tilde{\mathbf{R}} \Big), \tag{11}$$

where T is the target sequence length and θ denotes the trainable LoRA parameters. This encourages coherent generation under both retrieval-available and retrieval-absent conditions.

3.3 Retrieval Utility-Guided Reinforcement Fine-Tuning

SFT optimizes token-level likelihood and provides no explicit signal for whether retrieved contexts are useful. We therefore introduce RUG-RFT to optimize sentence-level quality while regulating retrieval reliance. RUG-RFT combines a *Quality Reward* with a *Retrieval-Utility Reward* under Group Relative Policy Optimization (GRPO) (Shao et al., 2024), which estimates advantages from intra-group reward statistics without a separate value model.

GRPO with group sampling. Given an input $(\mathbf{I}, \mathbf{Z}, \mathbf{R})$, the current policy π_θ samples N candidate outputs $\{O_1, \ldots, O_N\}$, each scored by $\mathcal{R}(O_i)$ to obtain r_i. GRPO standardizes rewards within the group to compute

$$A_i = \frac{r_i - \mathrm{mean}\{r_1, \ldots, r_N\}}{\mathrm{std}\{r_1, \ldots, r_N\}}. \tag{12}$$

Let $c_1 = \pi_\theta(O_i \mid \mathbf{I}, \mathbf{Z}, \mathbf{R}) / \pi_{\mathrm{old}}(O_i \mid \mathbf{I}, \mathbf{Z}, \mathbf{R})$ and $c_2 = \mathrm{clip}(c_1, 1 - \varepsilon, 1 + \varepsilon)$. The policy is updated by maximizing

$$\mathcal{J}_{\mathrm{GRPO}}(\theta) = \mathbb{E}\left[\frac{1}{N} \sum_{i=1}^{N} \min(c_1 A_i,\, c_2 A_i) - \beta\, D_{\mathrm{KL}}(\pi_\theta \| \pi_{\mathrm{ref}}) \right], \tag{13}$$

where the KL term regularizes deviation from the reference policy π_{ref}.

3.3.1 RETRIEVAL-UTILITY REWARD

We quantify the contribution of retrieval for a *fixed output* using teacher-forcing conditional likelihoods. Given $(\mathbf{I}, \mathbf{Z}, \mathbf{R})$, the current policy π_θ generates $Y = (y_1, \ldots, y_T)$ under the retrieval-augmented condition. We then score the same Y with retrieval using π_θ and without retrieval using a frozen reference policy π_{ref}:

$$\ell^{\text{with}}(Y) = \frac{1}{T} \sum_{t=1}^{T} \log p_\theta(y_t \mid y_{<t}, \mathbf{I}, \mathbf{Z}, \mathbf{R}) \,, \tag{14}$$

$$\ell^{\text{wo}}(Y) = \frac{1}{T} \sum_{t=1}^{T} \log p_{\text{ref}}(y_t \mid y_{<t}, \mathbf{I}, \mathbf{Z}) \,. \tag{15}$$

Here, π_{ref} is the frozen SFT checkpoint and is identical to the KL anchor in Eq. 13. Using π_{ref} for $\ell^{\text{wo}}(Y)$ prevents the policy from increasing the reward by artificially lowering its retrieval-free likelihood. We define the retrieval-induced likelihood gain as

$$\Delta\ell_{\text{ret}}(Y) = \ell^{\text{with}}(Y) - \ell^{\text{wo}}(Y), \tag{16}$$

where a positive value indicates that retrieval increases the model's confidence in Y.

3.3.2 QUALITY REWARD AND REWARD COMPOSITION

To align with task metrics while regulating retrieval reliance, we combine a sentence-level Quality Reward with the Retrieval-Utility Reward. Let $\mathcal{R}_{\text{qual}}(Y)$ denote a translation-quality score, such as BLEU or ROUGE, computed against the ground truth. We retain only positive retrieval utility:

$$\mathcal{R}^{+}_{\text{util}}(Y) = \max(0,\, \Delta\ell_{\text{ret}}(Y)) \,. \tag{17}$$

To balance the magnitudes of the two rewards, we introduce

$$\gamma = \frac{\widehat{\mathbb{E}}\big[|\mathcal{R}_{\text{qual}}|\big]}{\max\Big(\widehat{\mathbb{E}}\big[\mathcal{R}^{+}_{\text{util}}\big],\, \tau\Big)}, \tag{18}$$

where $\widehat{\mathbb{E}}[\cdot]$ is an exponential moving average over recent batches, τ prevents early instability, and γ is treated as stop-gradient. The total reward used in Eq. 12 is

$$\mathcal{R}_{\text{total}}(Y) = \mathcal{R}_{\text{qual}}(Y) + \alpha \cdot \gamma \cdot \mathcal{R}^{+}_{\text{util}}(Y), \tag{19}$$

where α controls the strength of the Retrieval-Utility Reward.

4 EXPERIMENT

4.1 EXPERIMENTAL SETUP

Datasets and metrics. We use CSL-News (Li et al., 2025b) and YouTube-ASL (Uthus et al., 2023) as large-scale pretraining datasets for Chinese Sign Language and American Sign Language, respectively. We then evaluate SignRAG on four downstream SLT benchmarks: CSL-Daily (Zhou et al., 2021), PHOENIX-2014T (Camgoz et al., 2018), How2Sign (Duarte et al., 2021), and OpenASL (Shi et al., 2022). Their train/validation/test splits are 18,401/1,077/1,176, 7,096/519/642, 31,085/1,739/2,348, and 96,476/966/975, respectively. We report BLEU-1/2/3/4 (Papineni et al., 2002) and ROUGE-L (Lin, 2004) following standard SLT evaluation protocols.

Retrieval gallery protocol. For each dataset, we build a separate train-only retrieval gallery from its training split, where each entry stores the encoded sign representation and paired translation. Validation and test samples are used only as queries and are never included in the gallery; during training, the current query instance is excluded to prevent self-retrieval. The details are shown in Appendix C.7. A detailed discussion of the efficiency, computational cost, and retrieval scalability of SignRAG is provided in Appendix E.

Table 1: Main results on four SLT benchmarks. "w/ gloss" and "w/o gloss" denote methods with and without gloss supervision, respectively. "Ext." indicates whether external sign-language datasets are used for pre-training. For the "w/o gloss" setting, methods with and without external pre-training data are separated and compared independently. Bold scores indicate the best results within each dataset, gloss setting, and external-data setting. "Venue" denotes the publication venue and year.

Dataset	Gloss	Method	Venue	Ext.	R-L↑	B@1↑	B@2↑	B@3↑	B@4↑
CSL-Daily	w/ gloss	SignBT (Zhou et al., 2021)	CVPR'21	✗	49.31	51.42	37.26	27.76	21.34
		SLTUNET (Zhang et al., 2023)	ICLR'23	✗	54.08	54.98	41.44	31.84	25.01
		TS-SLT (Chen et al., 2022c)	NeurIPS'22	✗	55.72	55.44	42.59	32.87	25.79
		CV-SLT (Zhao et al., 2024)	AAAI'24	✗	**57.06**	**58.29**	**45.15**	**35.77**	**28.94**
	w/o gloss	GFSLT-VLP (Zhou et al., 2023)	ICCV'23	✗	36.44	39.37	24.93	16.26	11.00
		Sign2GPT (Wong et al., 2024)	ICLR'24	✗	42.36	41.75	28.73	20.60	15.40
		VAP (Jiao et al., 2024)	ECCV'24	✗	48.56	49.99	–	–	20.85
		C^2RL (Chen et al., 2025)	TCSVT'25	✗	48.21	49.32	36.28	27.54	21.61
		RVLF (Rao et al., 2026)	Findings CVPR'26	✗	55.92	54.96	42.45	33.33	26.71
		SignRAG-SFT	–	✗	55.43	54.22	41.07	32.21	26.06
		SignRAG-RFT	–	✗	**56.36**	**56.60**	**43.87**	**34.43**	**27.73**
		Uni-Sign (Li et al., 2025b)	ICLR'25	✓	56.51	55.08	42.14	32.98	26.36
		Geo-Sign (Fish & Bowden, 2026)	NeurIPS'25	✓	57.97	55.89	–	–	27.42
		SignRAG-SFT	–	✓	60.20	59.39	46.94	37.95	31.22
		SignRAG-RFT	–	✓	**62.34**	**61.12**	**48.74**	**39.73**	**32.91**
PHOENIX-2014T	w/ gloss	SignBT (Zhou et al., 2021)	CVPR'21	✗	49.54	50.80	37.75	29.72	24.32
		TS-SLT (Chen et al., 2022c)	NeurIPS'22	✗	53.48	**54.90**	42.43	34.46	28.95
		CV-SLT (Zhao et al., 2024)	AAAI'24	✗	**54.33**	54.88	**42.68**	**34.79**	**29.27**
	w/o gloss	GFSLT-VLP (Zhou et al., 2023)	ICCV'23	✗	42.49	43.71	33.18	26.11	21.44
		Sign2GPT (Wong et al., 2024)	ICLR'24	✗	48.90	49.54	35.96	28.83	22.52
		FLa-LLM (Chen et al., 2024)	LREC-COLING'24	✗	45.27	46.29	35.33	28.03	23.09
		VAP (Jiao et al., 2024)	ECCV'24	✗	51.28	53.07	–	–	26.16
		C^2RL (Chen et al., 2025)	TCSVT'25	✗	50.96	52.81	40.20	32.20	26.75
		RVLF (Rao et al., 2026)	Findings CVPR'26	✗	52.56	53.60	41.20	33.24	27.86
		SignRAG-SFT	–	✗	52.14	53.05	40.88	32.68	27.30
		SignRAG-RFT	–	✗	**53.07**	**54.09**	**42.62**	**34.31**	**28.63**
How2Sign	w/o gloss	GloFE-VN (Lin et al., 2023)	ACL'23	✗	12.6	14.9	7.3	3.9	2.2
		OpenSLT (Tarrés et al., 2023)	CVPRW/WiCV'23	✗	–	34.0	19.3	12.2	8.0
		C^2RL (Chen et al., 2025)	TCSVT'25	✗	27.0	29.1	18.6	12.9	9.4
		FLa-LLM (Chen et al., 2024)	LREC-COLING'24	✗	27.8	29.8	19.0	13.3	9.7
		VAP (Jiao et al., 2024)	ECCV'24	✗	27.8	39.2	–	–	12.9
		RVLF (Rao et al., 2026)	Findings CVPR'26	✗	34.7	39.4	25.3	18.6	14.3
		SignRAG-SFT	–	✗	34.7	40.2	25.9	18.4	13.7
		SignRAG-RFT	–	✗	**36.2**	**42.9**	**28.0**	**20.3**	**15.2**
		YouTube-ASL (Uthus et al., 2023)	NeurIPS D&B'23	✓	–	37.8	24.1	16.9	12.4
		SignMusketeers (Gueuwou et al., 2025)	Findings ACL'25	✓	–	41.5	27.2	19.3	14.3
		Uni-Sign (Li et al., 2025b)	ICLR'25	✓	36.0	40.2	27.1	19.7	14.9
		SSVP-SLT (Rust et al., 2024)	ACL'24	✓	38.4	43.2	28.8	20.8	15.5
		SignRAG-SFT	–	✓	36.7	43.3	28.6	20.7	15.6
		SignRAG-RFT	–	✓	**38.5**	**44.7**	**30.3**	**22.1**	**16.7**
OpenASL	w/o gloss	OpenASL (Shi et al., 2022)	EMNLP'22	✗	21.02	20.92	12.08	8.59	6.72
		GloFE-VN (Lin et al., 2023)	ACL'23	✗	21.75	21.56	12.74	9.05	7.06
		C^2RL (Chen et al., 2025)	TCSVT'25	✗	31.36	31.46	21.85	16.58	13.21
		VAP (Jiao et al., 2024)	ECCV'24	✗	41.38	45.92	–	–	21.23
		RVLF (Rao et al., 2026)	Findings CVPR'26	✗	43.92	48.73	35.40	27.99	22.84
		SignRAG-SFT	–	✗	42.96	46.24	34.25	27.12	22.24
		SignRAG-RFT	–	✗	**43.87**	**49.14**	**36.58**	**29.01**	**23.61**
		Uni-Sign (Li et al., 2025b)	ICLR'25	✓	43.22	49.35	36.22	28.55	23.14
		SignRAG-SFT	–	✓	45.33	50.31	38.45	30.28	25.12
		SignRAG-RFT	–	✓	**46.91**	**51.56**	**39.52**	**32.06**	**26.62**

Implementation details. We adopt CoSign (Jiao et al., 2023) with a three-layer Transformer-based skeleton encoder and use Qwen3-8B as the decoder-only LLM across all datasets. All experiments are conducted on four NVIDIA RTX 4090D GPUs, each equipped with 48 GB of memory, using bfloat16 precision. The first and second pre-training stages are run for 200 and 50 epochs, respectively, followed by 10 epochs of SFT and 2 epochs of RFT. We use SGD with a learning rate of 0.01 in the first pre-training stage and AdamW in all subsequent stages. The learning rates for SFT and RFT are set to 1×10^{-4} and 1×10^{-5}, respectively, with a retrieval dropout probability of $p = 0.5$. Checkpoints are selected based on validation BLEU-4 scores. For the number of retrieved items, we set $K = 3$. Full implementation details are provided in Appendix C.

4.2 Main Results

Table 1 summarizes the main results. SignRAG outperforms existing state-of-the-art methods across all evaluated datasets. In particular, on CSL-Daily, SignRAG further surpasses gloss-based ap-

Table 2: Ablation studies on CSL-Daily. Components in (a) are added cumulatively. Reward variants in (b) start from the same retrieval-augmented SFT model.

Configuration	R-L↑	B@1↑	B@2↑	B@3↑	B@4↑
(a) Framework components					
SFT baseline	55.03	51.16	37.99	29.40	23.46
+ Pretraining	58.78	56.81	44.85	36.18	29.72
+ Retrieval	60.20	59.39	46.97	37.95	31.22
+ **RUG-RFT**	**62.34**	**61.12**	**48.74**	**39.73**	**32.91**
(b) Reward design					
Retrieval-augmented SFT	60.20	59.39	46.97	37.95	31.22
Quality Reward only	61.67	60.76	48.42	38.97	31.97
Retrieval-Utility Reward only	61.37	60.55	47.93	38.62	31.66
Both rewards	**62.34**	**61.12**	**48.74**	**39.73**	**32.91**

proaches, suggesting that large-scale pretraining on external sign language data, together with the capabilities of modern LLMs, can partially compensate for the absence of costly gloss annotations. Even without pretraining on additional data, SignRAG achieves state-of-the-art performance under the same setting. The training protocol follows RVLF (Rao et al., 2026). It is worth noting that, due to the lack of a large-scale German sign language pretraining dataset, all results on PHOENIX-2014T are obtained without pretraining on additional data. Compared with the Chinese datasets, the gains of SignRAG over existing state-of-the-art methods are more moderate on the English datasets. This discrepancy may be attributed to differences in dataset difficulty and evaluation protocols between Chinese and English benchmarks. In addition, the YouTube-ASL data used for pretraining are incomplete, which may limit the effectiveness of large-scale pretraining on English sign language translation. Further details are provided in Appendix D.4. For a fair comparison, we do not include methods that use substantially larger pretraining datasets or introduce modules specifically designed for English SLT datasets (Zhang et al., 2024; Jang et al., 2025).

4.3 Ablation Study

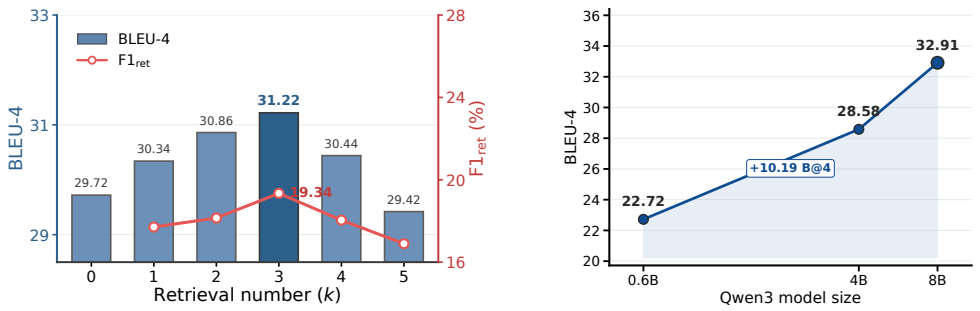
Table 2(a) evaluates the contributions of pretraining, retrieval augmentation, and RUG-RFT on CSL-Daily. Pretraining improves ROUGE-L from 55.03 to 58.78 and BLEU-4 from 23.46 to 29.72. Adding retrieval further raises them to 60.20 and 31.22, respectively. With RUG-RFT, performance reaches 62.34 ROUGE-L and 32.91 BLEU-4, with consistent gains across all metrics. These results demonstrate the effectiveness of progressively introducing pretraining, retrieval, and retrieval utility-guided reinforcement fine-tuning. Table 2(b) analyzes the reward design. Starting from the same retrieval-augmented SFT model, both the Quality Reward and Retrieval-Utility Reward improve performance independently. Their combination achieves the best results across all metrics, reaching 62.34 ROUGE-L and 32.91 BLEU-4. This confirms that translation quality and retrieval utility provide complementary optimization signals for RUG-RFT.

4.4 Analysis

Retrieval quality analysis. We investigate the relationship between the number of retrieved contexts, retrieval quality, and final translation performance. For each query, we vary the number of retrieved contexts K and measure the token-level overlap between the retrieved texts and the ground-truth translation. The resulting scores are then averaged over all queries to reflect the overall retrieval quality. Chinese sentences are segmented using Jieba, with punctuation, empty tokens, and symbol-only tokens removed. Let $\mathcal{T}_{\mathrm{GT}}$ denote the token set of the ground-truth translation, and let $\mathcal{T}_{\mathrm{ret}}^{(K)}$ denote the token set aggregated from the top-K retrieved translations. We define *Retrieval Recall*, *Retrieval Precision*, and *Retrieval F1* as

$$\mathrm{Rec}_{\mathrm{ret}}(K) = \frac{|\mathcal{T}_{\mathrm{GT}} \cap \mathcal{T}_{\mathrm{ret}}^{(K)}|}{|\mathcal{T}_{\mathrm{GT}}|}, \quad \mathrm{Prec}_{\mathrm{ret}}(K) = \frac{|\mathcal{T}_{\mathrm{GT}} \cap \mathcal{T}_{\mathrm{ret}}^{(K)}|}{|\mathcal{T}_{\mathrm{ret}}^{(K)}|}, \quad \mathrm{F1}_{\mathrm{ret}}(K) = \frac{2\,\mathrm{Rec}_{\mathrm{ret}}(K)\,\mathrm{Prec}_{\mathrm{ret}}(K)}{\mathrm{Rec}_{\mathrm{ret}}(K) + \mathrm{Prec}_{\mathrm{ret}}(K)}. \quad (20)$$

Here, $\mathrm{Rec}_{\mathrm{ret}}$ measures how much ground-truth lexical information is covered by the retrieved contexts, while $\mathrm{Prec}_{\mathrm{ret}}$ measures how much of the retrieved lexical content is relevant to the ground



(a) Retrieval quality vs. retrieval number (b) BLEU-4 vs. LLM size

Figure 3: Main analysis. **(a)** Relationship between retrieval quality and final translation performance under different numbers of retrieved contexts. **(b)** BLEU-4 scaling with LLM size. We evaluate Qwen3 backbones with different parameter scales (0.6B, 4B, and 8B) under the same training and evaluation protocol.

truth. Their harmonic mean, $\mathrm{F1}_{\mathrm{ret}}$, captures the trade-off between retrieval coverage and retrieval noise. As shown in Fig. 3a, as K increases, $\mathrm{F1}_{\mathrm{ret}}$ first improves and then continuously decreases, and the final translation performance exhibits the same trend. This is because introducing additional less-relevant contexts can instead interfere with SLT performance. All BLEU-4 scores in Fig. 3a are obtained from SFT models.

Model scaling analysis. We further investigate how the scale of the language model affects translation performance. As shown in Fig. 3b, under the same training and evaluation protocol, BLEU-4 increases from 22.72 to 28.58 and 32.91 when scaling the Qwen3 backbone from 0.6B to 4B and 8B, respectively. The consistent improvement shows that contemporary LLMs also exhibit stronger capabilities in sign language translation. These results further indicate that SignRAG can effectively benefit from scaling the underlying LLM.

Case study. We provide a qualitative comparison with the current SOTA model, Geo-Sign (Fish & Bowden, 2026), to analyze the effect of retrieved contexts. As shown in Table 3, the reference sentence contains fine-grained semantic cues, including "Traditionally" and "lucky money." Although Geo-Sign produces a partially related translation, it fails to preserve these cues. In contrast, SignRAG retrieves semantically aligned contexts that provide relevant topical and lexical cues. This further supports our finding that retrieval can improve translation performance. Additional qualitative examples are provided in Table 15 in the Appendix. Meanwhile, the bad case presented in Appendix D.7 shows that noisy retrieval can mislead the model, further motivating RUG-RFT.

Source	Sentence
Reference	红包在传统意义上也叫压岁钱。*(Traditionally, a red envelope is also called "lucky money.")*
Retrieved	[1] 你打算怎么花这笔压岁钱呢？*(How do you plan to spend this lucky money?)* [2] 带着祝福的红包比攀比中的计较更有意义。*(A red envelope carrying good wishes is more meaningful than one given for comparison or competition.)* [3] 你打算怎么花这笔压岁钱呢？*(How do you plan to spend this lucky money?)*
Geo-Sign	红包在历史上的原意是成年礼。*(Historically, the original meaning of a red envelope was a coming-of-age gift.)*
SignRAG	红包在传统意义上是压岁钱。*(Traditionally, a red envelope means "lucky money.")*

Table 3: Qualitative comparison on CSL-Daily.

5 Conclusion

We propose SignRAG, a unified framework for gloss-free sign language translation. SignRAG strengthens sign representation learning through staged pretraining and leverages target-domain semantic memory to provide instance-specific linguistic cues for downstream translation. Furthermore, RUG-RFT jointly considers translation quality and retrieval utility to encourage effective use of beneficial retrieved contexts. Experiments across multiple SLT benchmarks show that SignRAG achieves new state-of-the-art performance and surpasses existing gloss-supervised methods across all reported metrics on CSL-Daily. Future work will explore more efficient and robust retrieval mechanisms, as well as knowledge transfer across datasets and languages.

AI USE STATEMENT

We used generative AI tools primarily to assist with writing and language polishing, including improvements to grammar, wording, clarity, and readability. We did not use generative AI tools for research design, experimental design, data analysis, or result interpretation. All AI-assisted text was carefully reviewed and revised by the authors, who take full responsibility for the final content of the manuscript.

REPRODUCIBILITY STATEMENT

Although we have provided the experimental settings, training protocols, and implementation details as comprehensively as possible in the main paper and Appendix, we recognize that reproducibility is essential for impactful scientific research. To facilitate further verification and future research, upon acceptance, we will release the source code, pretrained checkpoints from different training stages, and models of different sizes. We hope these resources will support reproducible evaluation and further academic research.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

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.

Wenhu Chen, Hexiang Hu, Xi Chen, Pat Verga, and William W. Cohen. MuRAG: Multimodal retrieval-augmented generator for open question answering over images and text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5558–5570, 2022a.

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, 2022b.

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, 2022c.

Zhigang Chen, Benjia Zhou, Jun Li, Jun Wan, Zhen Lei, Ning Jiang, Quan Lu, and Guoqing Zhao. Factorized learning assisted with large language model for gloss-free sign language translation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pp. 7071–7081, 2024.

Zhigang Chen, Benjia Zhou, Yiqing Huang, Jun Wan, Yibo Hu, Hailin Shi, Yanyan Liang, Zhen Lei, and Du Zhang. C^2rl: Content and context representation learning for gloss-free sign language translation and retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025. doi: 10.1109/TCSVT.2025.3553052.

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.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-LLMs. *arXiv preprint arXiv:2406.07476*, 2024.

Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giró-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.

Edward Fish and Richard Bowden. Geo-sign: Hyperbolic contrastive regularisation for geometrically aware sign language translation. *Advances in Neural Information Processing Systems*, 38: 99293–99330, 2026.

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.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Shester Gueuwou, Xiaodan Du, Greg Shakhnarovich, and Karen Livescu. Signmusketeers: An efficient multi-stream approach for sign language translation at scale. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 22506–22521, 2025. doi: 10.18653/v1/2025.findings-acl.1157.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Jerry Huang, Siddarth Madala, Risham Sidhu, Cheng Niu, Hao Peng, Julia Hockenmaier, and Tong Zhang. RAG-RL: Advancing retrieval-augmented generation via reinforcement learning and curriculum learning. *arXiv preprint arXiv:2503.12759*, 2025.

Yu Huang, Junyang Lin, Chang Zhou, Hongxia Yang, and Longbo Huang. Modality competition: What makes joint training of multi-modal network fail in deep learning? (provably). In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 2022.

Youngjoon Jang, Liliane Momeni, Zifan Jiang, Joon Son Chung, Gül Varol, and Andrew Zisserman. Lost in translation, found in embeddings: Sign language translation and alignment. *arXiv preprint arXiv:2512.08040*, 2025.

Peiqi Jiao, Yuecong Min, Yanan Li, Xiaotao Wang, Lei Lei, and Xilin Chen. Cosign: Exploring co-occurrence signals in skeleton-based continuous sign language recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20676–20686, 2023.

Peiqi Jiao, Yuecong Min, and Xilin Chen. Visual alignment pre-training for sign language translation. In *Computer Vision – ECCV 2024*, pp. 349–367, 2024. doi: 10.1007/978-3-031-72946-1_20.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Dongxu Li, Chenchen Xu, Xin Yu, Kaihao Zhang, Benjamin Swift, Hanna Suominen, and Hongdong Li. Tspnet: Hierarchical feature learning via temporal semantic pyramid for sign language translation. *Advances in Neural Information Processing Systems*, 33:12034–12045, 2020.

Yuan Li, Qi Luo, Xiaonan Li, Bufan Li, Qinyuan Cheng, Bo Wang, Yining Zheng, Yuxin Wang, Zhangyue Yin, and Xipeng Qiu. R3-RAG: Learning step-by-step reasoning and retrieval for LLMs via reinforcement learning. *arXiv preprint arXiv:2505.23794*, 2025a.

Zecheng Li, Wengang Zhou, Weichao Zhao, Kepeng Wu, Hezhen Hu, and Houqiang Li. Uni-sign: Toward unified sign language understanding at scale. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=0Xt7uT04cQ.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, 2004.

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, 2023. doi: 10.18653/v1/2023.acl-long.722.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36:34892–34916, 2023.

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

Brandon McKinzie, Zhe Gan, Jean-Philippe Fatahalian, Alexander Toshev, Philipp Dufter, Alon Shalit, and Yinfei Yang. Mm1: Methods, analysis & insights from multimodal LLM pre-training. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

MMPose Contributors. OpenMMLab pose estimation toolbox and benchmark. `https://github.com/open-mmlab/mmpose`, 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, 2002.

Xiaokang Peng, Yake Wei, Andong Deng, Dong Wang, and Di Hu. Balanced multimodal learning via on-the-fly gradient modulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8238–8247, June 2022.

Maja Popović. chrF: Character n-gram f-score for automatic mt evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pp. 392–395, 2015.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763, 2021.

Zhi Rao, Yucheng Zhou, Benjia Zhou, Yiqing Huang, Sergio Escalera, and Jun Wan. Rvlf: A reinforcing vision-language framework for gloss-free sign language translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 9237–9247, June 2026.

Phillip Rust, Bowen Shi, Skyler Wang, Necati Cihan Camgöz, and Jean Maillard. Towards privacy-aware sign language translation at scale. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8624–8641, 2024. doi: 10.18653/v1/2024.acl-long.467.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Bowen Shi, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. Open-domain sign language translation learned from online video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 6365–6379, 2022.

Garrett Tanzer and Biao Zhang. Youtube-sl-25: A large-scale, open-domain multilingual sign language parallel corpus. In *International Conference on Learning Representations*, volume 2025, pp. 81921–81934, 2025.

Laia Tarrés, Gerard I. Gállego, Amanda Duarte, Jordi Torres, and Xavier Giró-i Nieto. Sign language translation from instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 5625–5635, 2023.

Dave Uthus, Garrett Tanzer, and Manfred Georg. Youtube-ASL: A large-scale, open-domain american sign language-english parallel corpus. *Advances in Neural Information Processing Systems*, 36:29029–29047, 2023.

Qiuchen Wang, Ruixue Ding, Yu Zeng, Zehui Chen, Lin Chen, Shihang Wang, Pengjun Xie, Fei Huang, and Feng Zhao. VRAG-RL: Empower vision-perception-based RAG for visually rich information understanding via iterative reasoning with reinforcement learning. *arXiv preprint arXiv:2505.22019*, 2025.

Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. Sign2GPT: Leveraging large language models for gloss-free sign language translation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=LqaEEs3UxU.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Huijie Yao, Wengang Zhou, Hao Zhou, Hezhen Hu, and Houqiang Li. Retrieval-augmented sign language translation. *ACM Transactions on Multimedia Computing, Communications and Applications*, 22(2), 2026. doi: 10.1145/3771277.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. Mplug-owi2: Revolutionizing multi-modal large language model with modality collaboration. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13040–13051. IEEE, 2024.

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.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Biao Zhang, Mathias Müller, and Rico Sennrich. SLTUNET: A simple unified model for sign language translation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=EBS4C77p_5S.

Biao Zhang, Garrett Tanzer, and Orhan Firat. Scaling sign language translation. *Advances in neural information processing systems*, 37:114018–114047, 2024.

Jian Zhao, Weizhen Qi, Wengang Zhou, Nan Duan, Ming Zhou, and Houqiang Li. Conditional sentence generation and cross-modal reranking for sign language translation. *IEEE Transactions on Multimedia*, 24:2662–2672, 2021.

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.

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

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.

Qidan Zhu, Jing Li, Fei Yuan, Jiaojiao Fan, and Quan Gan. A chinese continuous sign language dataset based on complex environments. *arXiv preprint arXiv:2409.11960*, 2024.

APPENDIX

A Extended Related Work **15**

A.1 Sign Language Translation 15

A.2 RL for Retrieval-Enhanced Reasoning 15

B Detailed Formulations and Derivations **16**

B.1 Vision-Language Pre-training 16

B.2 Attention-weighted Skeleton–Text Similarity 16

B.3 Bidirectional InfoNCE Loss 16

B.4 Memory-bounded distributed retrieval. 16

C More Implementation Details **17**

C.1 Computing Infrastructure 18

C.2 Keypoint Extraction. 18

C.3 Pretraining(Stage I): Sign-text pre-training. 19

C.4 Pretraining(Stage II): Large-scale sign MLLM pretraining. 19

C.5 SFT: Retrieval-Augmented Supervised Fine-Tuning. 19

C.6 RFT: Retrieval Utility-Guided Reinforcement Fine-Tuning. 20

C.7 Identical-text filtering in training retrieval. 20

D Further Analyses **21**

D.1 Analysis of different pretraining strategies. 21

D.2 Directly Fine-Tuning a Video LLM. 21

D.3 Why Does Hierarchical Pretraining Benefit Sign MLLMs? 22

D.4 Dataset statistics. 23

D.5 Metric overfitting analysis. 23

D.6 Transferability across Chinese SLT benchmarks. 24

D.7 Failure Case (Harmful Reliance on Retrieved Contexts). 24

E Efficiency, Computational Cost, and Retrieval Scalability **24**

E.1 Online SLT Inference Efficiency 24

E.2 Offline Retrieval Efficiency on CSL-Daily 25

E.3 Retrieval Scalability on OpenASL 26

E.4 Training Cost 26

E.5 Discussion on Million-Scale Retrieval 26

F More Cases **27**

A Extended Related Work

This section provides the full version of the related-work discussion that is condensed into a single paragraph in the main text.

A.1 Sign Language Translation

Sign language translation (SLT) aims to recover the semantic content conveyed by a signer from a sequence of performed signs. Unlike standard video understanding tasks, SLT must account for the unique linguistic structure of sign language. Specifically, sign sequences are more naturally aligned with gloss units, whereas spoken language sentences often follow substantially different grammatical patterns. This mismatch makes direct translation challenging. As a result, early SLT methods (Camgoz et al., 2020; Zhou et al., 2021; Chen et al., 2022c;b) commonly relied on gloss supervision, typically by introducing auxiliary Sign Language Recognition modules or explicitly modeling gloss generation before sentence translation. Despite their strong performance, such methods depend heavily on costly gloss annotations, which limits scalability. To address this issue, recent work has increasingly focused on gloss-free SLT, which learns a direct mapping from sign videos to spoken language sentences. Early studies (Camgoz et al., 2018; 2020; Li et al., 2020; Zhao et al., 2021; Yin et al., 2023) demonstrated the feasibility of this setting, although a clear performance gap from gloss-based methods remains. More recent efforts attempt to close this gap by improving visual-text representation learning. For example, GFSLT (Zhou et al., 2023) adapts CLIP (Radford et al., 2021) to sign language, inspiring a series of vision-language SLT models (Wong et al., 2024; Chen et al., 2024; Gong et al., 2024; Jiao et al., 2024; Chen et al., 2025). Another line of work improves SLT from the data perspective. Recent studies (Uthus et al., 2023; Rust et al., 2024; Li et al., 2025b; Zhang et al., 2024; Gueuwou et al., 2025; Jang et al., 2025; Fish & Bowden, 2026) show that pre-training on large-scale sign corpora can significantly benefit downstream translation. However, most existing efforts focus on pre-training T5-based encoder–decoder architectures, whose language priors, general knowledge, and generative capabilities remain limited compared with contemporary LLMs. Therefore, how to design effective pre-training strategies for adapting sign representations to contemporary LLM-based SLT models remains largely unexplored.

A.2 RL for Retrieval-Enhanced Reasoning

Retrieval-augmented generation (RAG) enhances parametric language models by incorporating non-parametric external contexts, where retrieved evidence is used as additional information during generation (Lewis et al., 2020; Chen et al., 2022a). By providing explicit external references, RAG improves knowledge utilization and reduces the dependence on purely parametric memory. Beyond general text generation, retrieval-based approaches have also been explored in sign language translation. However, existing retrieval-based SLT methods (Yao et al., 2026) mainly focus on gloss-based settings, where retrieval is performed under the guidance of gloss representations.

In parallel, reinforcement learning (RL) has become an effective post-training strategy for improving the reasoning and generation abilities of large language models. Group Relative Policy Optimization (GRPO) (Shao et al., 2024; Guo et al., 2025) provides an efficient RL framework by estimating relative advantages among grouped samples without requiring an additional value model. Recent studies have explored combining RL with RAG to optimize retrieval-enhanced reasoning and generation. RAG-RL (Huang et al., 2025) improves answer generation in retrieval contexts through reinforcement learning and curriculum strategies. R3-RAG (Li et al., 2025a) further enables joint reasoning and retrieval optimization with outcome and process rewards. VRAG-RL (Wang et al., 2025) extends this paradigm to visual-rich scenarios by jointly optimizing retrieval, reasoning, and generation.

These studies demonstrate the potential of RL for dynamically optimizing the interaction between retrieval and generation. However, existing RL-based RAG methods mainly target knowledge-intensive reasoning tasks, rather than multimodal translation with sign language inputs. In contrast, SignRAG investigates how large-scale sign pretraining should be organized under a modern LLM decoder, and complements parametric adaptation with a sign-to-text retrieval gallery and retrieval utility-guided reinforcement fine-tuning. Unlike previous retrieval-based SLT approaches,

SignRAG operates without gloss annotations and explicitly optimizes the usefulness of retrieved translation contexts during generation.

B Detailed Formulations and Derivations

B.1 Vision-Language Pre-training

For completeness, we provide additional details of the vision-language pre-training procedure.

B.2 Attention-weighted Skeleton–Text Similarity

Skeleton and text features are projected into a shared D-dimensional space and ℓ_2-normalized. Given skeleton features $\{f_i\}_{i=1}^{M}$ and text features $\{t_j\}_{j=1}^{L}$, we compute token-level similarities

$$E_{i,j} = f_i^\top t_j. \tag{21}$$

To calculate the global skeleton-to-text similarity $z_{S,T}$, a row-wise Softmax (Cheng et al., 2023) is applied to $\boldsymbol{E}$ to obtain the attention weights $\boldsymbol{P}$:

$$P_{i,j} = \frac{\exp(E_{i,j}/\tau)}{\sum_{k=1}^{L} \exp(E_{i,k}/\tau)}, \tag{22}$$

where τ denotes a trainable temperature parameter. The global skeleton–text similarity is defined as

$$z_{S,T} = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{L} P_{i,j}\, E_{i,j}. \tag{23}$$

The text-to-skeleton similarity is computed analogously.

B.3 Bidirectional InfoNCE Loss

Given a mini-batch of B paired samples, we construct batch-level similarity matrices and apply a bidirectional InfoNCE objective with temperature τ'.

B.4 Memory-bounded distributed retrieval.

This section describes the implementation of memory-efficient retrieval in SignRAG. Let $\mathcal{G}$ denote a training gallery containing N sign sequences, each represented by frame-level features with channel dimension C. Loading the entire gallery into GPU memory requires storage proportional to $\mathcal{O}(NLC)$, where L denotes the padded temporal length. Comparing a query batch against the full gallery further introduces memory costs for pairwise similarities and intermediate computations. To support large galleries, we combine CPU-resident gallery storage, distributed sharding, and block-wise similarity computation, followed by exact distributed top-K aggregation.

CPU-resident gallery sharding. With R distributed workers, we partition the gallery into disjoint shards:

$$\mathcal{G} = \bigcup_{r=1}^{R} \mathcal{G}^{(r)}, \qquad \mathcal{G}^{(r)} \cap \mathcal{G}^{(r')} = \varnothing, \quad r \neq r'. \tag{24}$$

Each worker is responsible for approximately N/R gallery samples. The encoded features of its local shard are retained in CPU memory and transferred to GPU only for the current similarity computation. Thus, no worker needs to keep its entire gallery shard, or the complete gallery, in GPU memory.

Block-wise similarity computation. We divide the queries into chunks $\mathcal{Q}_b$ of at most B_q sequences and each local gallery shard into chunks $\mathcal{G}_c^{(r)}$ of at most B_g sequences:

$$|\mathcal{Q}_b| \le B_q, \qquad |\mathcal{G}_c^{(r)}| \le B_g. \tag{25}$$

Each query chunk is broadcast to all workers. Worker r then compares it against the chunks of its local gallery shard in turn. For the current query–gallery block, the aggregated similarities are

$$A_{b,c}^{(r)}[u,v] = S\left(Q_u^b, G_v^{(r,c)}\right), \qquad A_{b,c}^{(r)} \in \mathbb{R}^{|\mathcal{Q}_b|\times|\mathcal{G}_c^{(r)}|}, \tag{26}$$

where $S(\cdot,\cdot)$ is the attention-weighted sign–sign similarity defined in Eq. 8. After evaluating a block, its similarity scores are transferred to CPU memory, and the GPU storage occupied by the current gallery chunk can be released before loading the next one. The query chunk remains available while the worker processes its local gallery. Scores from successive gallery chunks are collected on CPU for local candidate selection.

GPU memory usage. Let L bound the padded temporal lengths of the active query and gallery chunks. The resident retrieval tensors on each GPU require

$$\mathcal{M}_{\text{resident}} = \mathcal{O}\left((B_q + B_g)LC + B_qB_g\right). \tag{27}$$

The first term accounts for the active query and gallery features, and the second accounts for their aggregated similarity scores. This bound excludes model parameters, temporary intermediate activations, and kernel workspace; the actual peak GPU memory also depends on the implementation of $S(\cdot,\cdot)$. For fixed chunk sizes and sequence lengths, the resident GPU storage is independent of the total gallery size N. The chunk sizes can therefore be adjusted to the available device memory. CPU storage still grows with the local gallery size and the similarity scores retained for candidate selection, and retrieval still evaluates every eligible query–gallery pair.

Candidate filtering. During training-set retrieval, we exclude the query itself, identified by its sample identity, and gallery samples whose reference translations are identical to that of the query. These exclusions prevent trivial retrieval of the query or entries with identical supervision. Let y_x denote the reference translation of sample x. The eligible local gallery for a training query q is

$$\mathcal{E}^{(r)}(q) = \left\{ g \in \mathcal{G}^{(r)} \;\middle|\; g \neq q,\ y_g \neq y_q \right\}. \tag{28}$$

For development and test queries, retrieval is performed against the training gallery without using the query reference translation, so $\mathcal{E}^{(r)}(q) = \mathcal{G}^{(r)}$. All exclusions are applied before local top-K selection to ensure that eligible candidates are not displaced by entries that would subsequently be removed.

Distributed top-K aggregation. After evaluating its local gallery, worker r selects

$$\mathcal{C}_K^{(r)}(q) = \mathrm{TopK}_{g\in\mathcal{E}^{(r)}(q)}\, S(q,g), \tag{29}$$

returning all eligible entries if fewer than K are available. Only the selected sample identifiers and their similarity scores are communicated for final aggregation. The global retrieved set is

$$\mathcal{R}_K(q) = \mathrm{TopK}_{g\in\bigcup_{r=1}^{R}\mathcal{C}_K^{(r)}(q)}\, S(q,g). \tag{30}$$

This aggregation preserves exact exhaustive retrieval. Under a consistent ordering for tied scores, any candidate outside its shard's local top-K is preceded by at least K eligible candidates from that shard and therefore cannot belong to the global top-K. Consequently, selecting from the union of local candidates yields the same result as selecting directly from the full eligible gallery. Final aggregation requires at most RK candidates per query, without gathering the gallery features or the full similarity matrix across workers.

Communication and CPU fallback. In our implementation, variable-length query payloads and retrieved candidate metadata are communicated through a CPU-side Gloo process group. This avoids the additional GPU-memory allocation associated with object collectives over NCCL. If a query–gallery block still exceeds the available GPU memory, the implementation supports a CPU fallback that evaluates the same similarity function $S(\cdot,\cdot)$. The fallback changes the execution device while preserving the scoring definition, eligibility rules, and exhaustive retrieval procedure.

C More Implementation Details

To enable a thorough understanding of SignRAG, this section presents more implementation details.

Table 4: Implementation details of the four training stages. LR denotes learning rate, FFN denotes feed-forward network, and RFT denotes reinforcement fine-tuning.

Stage	Optimization	Model Configuration	Data / Training Details
Pretraining I *Sign–Text Pretraining*	**Epochs:** 200 **Optimizer:** SGD **LR:** 0.01 **Weight decay:** 0.001 **Scheduler:** cosine **Warmup:** 20 epochs **Batch size:** 8	**Text encoder/decoder:** first three layers of mBART **Skeleton encoder:** three-layer Transformer **Attention heads:** 8 **Hidden size:** 512 **FFN dimension:** 2048 **Contrastive dimension:** 1024	**Training corpus:** CSL-News **Vocabulary size:** 11,540
Pretraining II *Large-Scale Sign MLLM Pretraining*	**Epochs:** 50 **Optimizer:** AdamW **LR:** 1×10^{-4} **Batch size:** 8 **Objective:** cross-entropy	**Base LLM:** Qwen3-8B **Projector:** MLP with GELU **Projection:** $512 \rightarrow 4096$ **LoRA targets:** `q_proj`, `v_proj` **LoRA rank:** 16 **Scaling factor:** 32 **Dropout:** 0.3	**Training corpora:** CSL-News / YouTube-ASL The sign encoder, multimodal projector, and LLM are jointly pretrained.
SFT *Retrieval-Augmented SFT*	**Epochs:** 10 **Optimizer:** AdamW **LR:** 1×10^{-4} **Batch size:** 4 **Objective:** cross-entropy	Stage-II LoRA weights are merged into the base LLM, followed by a new set of LoRA modules. **LoRA targets:** `q_proj`, `v_proj` **LoRA rank:** 16 **Scaling factor:** 32 **Dropout:** 0.3	**Training data:** downstream SLT datasets Retrieved texts and sign features are jointly used as multimodal input. **Prompt:** Fig. 4
RFT *Retrieval Utility-Guided RFT*	**Epochs:** 2 **Optimizer:** AdamW **LR:** 1×10^{-5} **Batch size:** 8 **Responses per input:** 8	SFT LoRA weights are merged with the pretrained LLM weights, followed by a new set of LoRA modules. **LoRA targets:** `q_proj`, `v_proj` **LoRA rank:** 64 **Scaling factor:** 128 **Dropout:** 0.05	**Optimization:** GRPO **Rewards:** sentence-level quality reward and retrieval-utility reward The same prompt template as SFT is used.

C.1 Computing Infrastructure

The entire experimental process was conducted on a computational infrastructure comprising an AMD EPYC 7502 32-Core Processor, 4 NVIDIA RTX 4090D GPUs with 48GB of VRAM, running the Ubuntu 20.04 operating system, with Python 3.10, CUDA 11.8, transformers 4.51.3 and PyTorch 2.1.2.

C.2 Keypoint Extraction.

Following the preprocessing protocol adopted in CoSign (Jiao et al., 2023), we extract frame-level human keypoints from sign-language videos using the off-the-shelf MMPose estimator. To focus on motion and appearance cues that are most relevant to sign understanding, we retain a compact set of keypoints covering the upper body, hands, mouth, and face. Specifically, the selected keypoints include 9 body joints, 21 joints for each hand, 8 mouth landmarks, and 18 facial landmarks, resulting in 77 keypoints per frame. Additional implementation details follow CoSign (Jiao et al., 2023).

C.3 PRETRAINING(STAGE I): SIGN-TEXT PRE-TRAINING.

In the first stage, we train a compact sign-text model to align skeleton-based sign representations with textual semantics. Following GFSLT-VLP (Zhou et al., 2023), the text encoder and decoder are initialized by reusing the first three layers of mBART, while the skeleton encoder adopts the same three-layer Transformer structure. Each Transformer block uses 8 attention heads, a hidden size of 512, and a feed-forward dimension of 2048. The model is optimized for 200 epochs using SGD with a learning rate of 0.01 and a weight decay of 0.001. A cosine learning rate schedule is applied with 20 warmup epochs. The batch size is set to 8. For CSL-News, we follow prior work and reduce the vocabulary to a task-specific subset of 11,540 tokens. The contrastive feature dimension for both visual and textual representations is set to 1024.

C.4 PRETRAINING(STAGE II): LARGE-SCALE SIGN MLLM PRETRAINING.

We use Qwen3-8B as the base language model. Following the design of LLaVA (Liu et al., 2023), an MLP projector with GELU activation maps the 512-dimensional visual features into the 4096-dimensional embedding space of the LLM. To enable parameter-efficient adaptation, we apply LoRA to the query and value projection matrices in the self-attention layers of the LLM. The LoRA rank, scaling parameter, and dropout rate are set to 16, 32, and 0.3, respectively. The model is trained for 50 epochs using the AdamW optimizer with a learning rate of 1×10^{-4} and a batch size of 8. Cross-entropy loss is used as the training objective.

C.5 SFT: RETRIEVAL-AUGMENTED SUPERVISED FINE-TUNING.

After large-scale sign MLLM pretraining, we merge the learned LoRA parameters into the base LLM to obtain the final pretrained LLM checkpoint. We then attach a new set of LoRA modules using the same LoRA configuration, including the rank, scaling factor, and dropout rate. The main difference in this stage is that the training data are switched from large-scale pretraining corpora, such as CSL-News and YouTube-ASL, to downstream SLT datasets. Retrieved texts are additionally incorporated and combined with the sign features to construct the multimodal input. The system prompt and instruction template used in this stage are shown in Fig. 4. During SFT, the model is trained for 10 epochs using AdamW with a learning rate of 1×10^{-4}. The batch size is set to 4, and cross-entropy loss is used as the training objective.

(a) System Prompt

```
A conversation between a deaf person and an artificial intelligence assistant. The assistant
generates accurate and fluent translation text based on the provided sign language video
information.
```

(b) Instruction (with RAG)

```
<image>This is sign language video information.
Retrieved texts (reference evidence, may not be entirely accurate):
{ref_texts}
Please generate the target translation based on the video; you may refer to
keywords/expressions from the retrieved texts, but do not directly copy uncertain content.
```

(c) Instruction (without RAG)

```
<image>This is sign language video information.
No retrieved texts available.
Please generate the target translation based solely on the sign language video information.
```

Figure 4: System prompt and instruction template. Here, *with RAG* denotes the setting where retrieved contexts are provided to the LLM, while *without RAG* denotes the setting where no retrieved contexts are provided.

Algorithm 1 Retrieval Context Construction

Input: Query q with id_q and t_q; gallery $\mathcal{G} = \{(g_i, t_i, id_i)\}_{i=1}^{N}$; similarity function $S(\cdot, \cdot)$; retrieval number K
Output: Retrieved contexts $\mathcal{R}_K(q)$

Compute visual similarities:
$s_i = S(q, g_i), \qquad i = 1, \ldots, N$
Rank gallery samples in descending order:
$\pi = \mathrm{Argsort}_i(s_i)$
Initialize $\mathcal{R}_K(q) \leftarrow \varnothing$
for $i \in \pi$ **do**
  if $id_i = id_q$ **then**
    continue
  end if
  if TRAINRETRIEVAL $\wedge\ t_i = t_q$ **then**
    continue
  end if
  $\mathcal{R}_K(q) \leftarrow \mathcal{R}_K(q) \cup \{t_i\}$
  if $|\mathcal{R}_K(q)| = K$ **then**
    break
  end if
end for
return $\mathcal{R}_K(q)$

C.6 RFT: RETRIEVAL UTILITY-GUIDED REINFORCEMENT FINE-TUNING.

After supervised fine-tuning, we merge the learned LoRA parameters with our pretrained LLM weights to obtain the SignRAG-SFT checkpoint. We then attach a new set of LoRA modules for reinforcement fine-tuning. In this stage, the LoRA rank is increased to 64, the scaling factor is set to 128, and the dropout rate is reduced to 0.05. The model is trained for 2 epochs using AdamW with a learning rate of 1×10^{-5} and a batch size of 8. Following GRPO, we sample 8 candidate responses for each input and compute group-relative advantages based on both the sentence-level quality reward and the retrieval-utility reward. The same prompt template used in the SFT stage is adopted for RFT. For datasets containing very short sentences, we append sentence-final punctuation before reward computation to reduce the instability of BLEU-4-based training signals.

C.7 IDENTICAL-TEXT FILTERING IN TRAINING RETRIEVAL.

We use CSL-Daily as an example to illustrate the retrieval procedure. The retrieval gallery $\mathcal{G}$ consists of all training samples, where each gallery entry is represented as $\mathcal{G}_i = (g_i, t_i, id_i)$, containing the sign sample, its translation, and sample ID. Given a query q, all gallery samples are first ranked according to their visual similarity, followed by sample-level and text-level filtering. The complete procedure is summarized in Algorithm 1.

During training-set self-retrieval, identical-text filtering is introduced to prevent the target sentence of the query from being directly provided as retrieval context. Without this filtering, samples sharing the same target annotation may dominate the retrieved candidates, allowing the translation model to exploit a trivial shortcut by copying the retrieved target sentence. More importantly, this would introduce a mismatch between training and dev/test inference, since the exact target translation of a query is generally unavailable in the retrieval gallery at inference time.The effectiveness of identical-text filtering on CSL-Daily is evaluated in Table 5. Only 2 test samples share identical translations with the training set, suggesting that the improvements of SignRAG are not driven by a trivial retrieval shortcut that directly matches test queries to training samples with the same target translation.

Table 5: Effect of identical-text filtering during training-set self-retrieval on CSL-Daily.

Setting	BLEU-4
SignRAG-SFT w/o identical-text filtering	11.21
SignRAG-SFT w/ identical-text filtering	31.22

D FURTHER ANALYSES

D.1 ANALYSIS OF DIFFERENT PRETRAINING STRATEGIES.

We compare three pretraining strategies. *Encoder-only pretraining* initializes only the sign encoder from Stage I before downstream SFT. *End-to-end pretraining* directly pretrains a randomly initialized sign encoder jointly with the LLM. *Hierarchical pretraining* first learns a language-aware sign encoder and then jointly pretrains the sign encoder, multimodal projector, and LLM. As shown in Table 6, hierarchical pretraining substantially outperforms direct end-to-end pretraining on CSL-Daily, achieving BLEU-4 and ROUGE-L scores of 29.72 and 58.78, compared with 9.53 and 38.70, respectively. These results highlight the importance of learning semantically grounded sign representations before integrating them with an LLM.

Table 6: Comparison of pretraining strategies. Results on CSL-News and CSL-Daily correspond to pretraining and downstream SFT evaluation, respectively.

Pretraining Strategy	**R-L↑**	**B@1↑**	**B@2↑**	**B@3↑**	**B@4↑**
Pretraining evaluation on CSL-News					
Sign Encoder + Shallow Transformer	36.31	41.58	30.24	23.51	19.02
Downstream SFT evaluation on CSL-Daily					
Encoder-only Pretraining	22.51	18.22	8.76	5.02	3.20
End-to-End Pretraining	38.70	32.74	19.87	13.30	9.53
Hierarchical Pretraining (Ours)	**58.78**	**56.81**	**44.85**	**36.18**	**29.72**

D.2 DIRECTLY FINE-TUNING A VIDEO LLM.

Constructing large-scale sign language datasets remains challenging due to the high cost of data collection and annotation, as well as the limited availability of annotators with professional sign language expertise. Consequently, a substantial cross-modal semantic gap persists between sign videos and natural language text. Video LLMs have demonstrated strong capabilities in temporal modeling and natural language generation. Adapting such models with a limited number of sign–text pairs therefore provides a potentially effective way to leverage their existing video understanding and language modeling capabilities for cross-modal alignment, while reducing the dependence on large-scale sign language annotations and costly training. From this perspective, adapting general-purpose Video LLMs to sign language translation represents a promising research direction.

To ensure a fair comparison and isolate the effect of the model architecture as much as possible, we strictly follow the official training configuration of VideoLLaMA2, keeping all training settings and hyperparameters unchanged while replacing only the original video data with sign language data. As shown in Fig. 5, our preliminary experiments reveal a substantial performance gap between directly fine-tuning VideoLLaMA2 (7B) (Cheng et al., 2024) on sign–text pairs and our SFT model. We attribute this gap in part to the distinctive nature of sign language, whose semantics are substantially more structured and abstract than those of general video content. The visual encoder of a general-purpose Video LLM is not specifically designed to model the fine-grained visual patterns underlying sign language and therefore struggles to learn sufficiently discriminative and linguistically relevant sign representations. Moreover, compared with generic video understanding, SLT requires visual semantics at a much finer granularity, demanding accurate modeling of handshape, motion, spatial location, non-manual cues, and their temporal compositions. Capturing such subtle linguistic information remains challenging for general-purpose video encoders.

These observations also motivate a central goal of our work: to move beyond the direct adaptation of existing general-purpose Video LLMs and toward a multimodal foundation model with stronger sign representation learning and language modeling capabilities for SLT. Although the current analysis is based on a relatively limited experimental setting, further exploring Video LLMs with stronger visual backbones, larger model scales, and more extensive sign-language-oriented pre-training remains an important direction for future work.

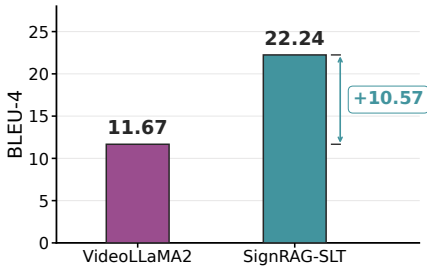


Figure 5: Performance comparison between directly fine-tuning VideoLLaMA2 and the SignRAG-SFT model.

D.3 Why Does Hierarchical Pretraining Benefit Sign MLLMs?

We further analyze why hierarchical pretraining is particularly important when replacing conventional pretrained language models with stronger LLMs. Empirically, we observe an interesting phenomenon: directly coupling a sign encoder with a powerful LLM can perform substantially worse than using a T5-based architecture, whereas pretraining the sign encoder before multimodal joint training significantly improves performance.

We attribute this phenomenon to **Cross-Modal Optimization Imbalance** (Wang et al., 2020; Huang et al., 2022; Peng et al., 2022). At the beginning of end-to-end multimodal training, the sign encoder is either randomly initialized or only weakly aligned with linguistic semantics, while the LLM already possesses strong language modeling capability and mature linguistic priors. Therefore, the sign representation

$$Z_s = E_{\text{sign}}(X) \tag{31}$$

may still mainly encode low-level motion patterns, whereas the LLM can already partially fit the target sentence through its pretrained language knowledge. This imbalance can lead to

$$\underbrace{\text{Weak Sign Features}}_{\text{visual pathway}} + \underbrace{\text{Strong Language Prior}}_{\text{LLM}} \rightarrow \text{Language-Dominated Optimization.} \tag{32}$$

As a result, the model may reduce the generation loss without sufficiently improving the semantic quality of the sign representation.

There are two main reasons for this behavior. First, the sign encoder only receives indirect supervision from the final generation objective:

$$\mathcal{L}_{\text{gen}} \rightarrow \text{LLM} \rightarrow \text{Projector} \rightarrow \text{Sign Encoder}. \tag{33}$$

Hence, the gradient signal reaching the sign encoder is less direct than explicit sign–text semantic supervision. Second, the multimodal projector may partially compensate for immature sign representations by mapping them into the LLM latent space, rather than forcing the sign encoder itself to develop linguistically meaningful features. Consequently, end-to-end optimization may successfully reduce the generation loss while still producing weak visual grounding. In other words,

$$\text{Lower Generation Loss} \not\Rightarrow \text{Better Sign Representation}. \tag{34}$$

This issue is less pronounced for T5-based SLT models. Compared with modern LLMs, T5-style language models generally provide weaker linguistic priors and therefore cannot rely as heavily on language-side shortcuts. To minimize the translation objective, they must depend more strongly on visual evidence from the sign encoder, resulting in tighter visual-to-language coupling. This explains why a T5-based model can remain relatively stable even when the sign encoder and language model are trained jointly.

In contrast, stronger LLMs require more semantically mature modality representations before multimodal optimization. Our hierarchical pretraining explicitly addresses this requirement. In the first stage, we perform sign–text alignment:

$$E_{\text{sign}} \xrightarrow{\text{Sign–Text Alignment}} Z_s^{\text{sem}}, \tag{35}$$

where Z_s^{sem} is encouraged to encode linguistically structured sign semantics rather than only motion patterns. In the second stage, the aligned sign encoder is integrated with the multimodal projector and LLM:

$$Z_s^{\text{sem}} \rightarrow \text{Projector} \rightarrow \text{LLM}. \tag{36}$$

This decomposition transforms a difficult joint optimization problem into two progressively learned objectives:

$$\boxed{\text{Sign Representation Learning} \rightarrow \text{Cross-Modal Generation}} \tag{37}$$

The first stage focuses on *what the sign means*, while the second stage focuses on *how to express the acquired sign semantics in natural language*.

Overall, these results suggest that the advantage of a stronger LLM can only be fully realized when it is paired with a sufficiently strong modality interface. More generally, effective Sign MLLM pretraining depends not only on language-model capacity, but also on the quality of modality representations, cross-modal alignment, and optimization order:

$$\boxed{\text{MLLM Performance} \propto \text{Representation Quality} + \text{Alignment Quality} + \text{Optimization Strategy}.} \tag{38}$$

This observation motivates our hierarchical design and explains why directly scaling the language backbone does not necessarily lead to better sign language translation.

D.4 Dataset statistics.

Table 7: Statistics of the pretraining and fine-tuning datasets. For YouTube-ASL, we report both the original dataset size and the number of samples available to us.

Stage	Dataset	Sign language	# Samples
Pretraining	CSL-News (Li et al., 2025b)	Chinese	722,715
	YouTube-ASL (Uthus et al., 2023) (original)	American	530,161
	YouTube-ASL (available)	American	434,953
Fine-tuning	How2Sign (Duarte et al., 2021)	American	35,172
	OpenASL (Shi et al., 2022)	American	98,417
	CSL-Daily (Zhou et al., 2021)	Chinese	20,654
	PHOENIX-2014T(Camgoz et al., 2018)	German	8,257

Table 7 summarizes the datasets used for pretraining and fine-tuning, all of which consist of sentence-level sign–text pairs. For YouTube-ASL, we successfully obtained 434,953 of the 530,161 reported samples, corresponding to 82.04% of the original dataset. Some of the provided YouTube links have become invalid over time, preventing access to the corresponding source videos. We therefore use the available subset for pretraining.

D.5 Metric overfitting analysis.

Since RUG-RFT optimizes sentence-level BLEU and ROUGE rewards, we further evaluate SignRAG with chrF (Popović, 2015) and BERTSim, a BERT-based semantic similarity score computed with *bert-base-chinese*.[1] As shown in Table 8, SignRAG-RFT improves over SignRAG-SFT on ROUGE-L, BLEU-4, chrF, and BERTSim. These complementary metrics suggest that the gains are not limited to direct BLEU/ROUGE-oriented optimization.

Table 8: Additional automatic metrics on CSL-Daily for checking metric overfitting. SFT and RFT denote SignRAG-SFT and SignRAG-RFT; R-L, B@4, and BSim denote ROUGE-L, BLEU-4, and BERTSim.

	R-L↑	**B@4↑**	**chrF↑**	**BSim↑**
SFT	60.20	31.22	26.93	82.10
RFT	**62.34**	**32.91**	**28.42**	**82.96**

[1] https://huggingface.co/google-bert/bert-base-chinese

D.6 TRANSFERABILITY ACROSS CHINESE SLT BENCHMARKS.

To further evaluate whether our pretraining strategy generalizes beyond a specific Chinese SLT dataset, we transfer the pretrained model to CE-CSL (Zhu et al., 2024), a continuous sign language benchmark collected under complex environments. As shown in Table 9, our `Pretrain→SFT` consistently outperforms Uni-Sign and Geo-Sign across all metrics. In particular, it achieves 24.44 BLEU-4 and 54.55 ROUGE-L, surpassing the strongest baselines by 2.37 and 2.54 points, respectively. These results demonstrate that the learned representations can effectively transfer across different Chinese SLT benchmarks, supporting the generalizability of our pretraining strategy.

Table 9: **Comparison on the CE-CSL benchmark.** We further evaluate the adaptability of different pretrained SLT models on CE-CSL. Uni-Sign and Geo-Sign are included as representative baselines with publicly available pretrained weights. The results demonstrate the capability of our `Pretrain→SFT` to transfer and adapt to different downstream SLT benchmarks.

Method	B@1↑	B@2↑	B@3↑	B@4↑	R-L↑
Uni-Sign	51.23	37.14	27.89	21.66	52.01
Geo-Sign	51.81	37.22	28.31	22.07	50.24
`Pretrain→SFT` (Ours)	**53.18**	**39.14**	**30.27**	**24.44**	**54.55**

D.7 FAILURE CASE (HARMFUL RELIANCE ON RETRIEVED CONTEXTS).

Table 10 presents a representative failure case illustrating harmful reliance on retrieved contexts. Although the `Pretrain→SFT` variant, which is fine-tuned using only visual input without retrieval, correctly translates the input as *The weather is nice today*, SignRAG-SFT is misled by the incorrect retrieved context and generates *The supermarket is holding an event*. After RFT, SignRAG-RFT recovers the correct translation, indicating that retrieval utility-guided reinforcement fine-tuning can reduce harmful reliance on misleading retrieved information.

Table 10: Qualitative example illustrating harmful reliance on retrieved contexts and the effect of RFT. Red text highlights misleading or incorrectly generated content.

Source	Text
Ground-Truth	The weather is nice today.
Retrieved Text	The supermarket is closed today.
`Pretrain→SFT`	The weather is nice today.
SignRAG-SFT	The supermarket is holding an event.
SignRAG-RFT	The weather is nice today.

E EFFICIENCY, COMPUTATIONAL COST, AND RETRIEVAL SCALABILITY

In this section, we provide a systematic analysis of the computational overhead, retrieval scalability, memory consumption, and training cost of SignRAG. Unless otherwise specified, all efficiency experiments are conducted using Qwen3-8B on four NVIDIA RTX 4090D GPUs. For a fair comparison, the Non-RAG baseline (`Pretrain→SFT`) and SignRAG-SFT are evaluated using the same hardware, batch size, numerical precision, and decoding strategy. Since inference latency reported in prior work is often measured under different hardware and implementation settings, we focus on the incremental cost of SignRAG relative to this hardware-controlled baseline.

E.1 ONLINE SLT INFERENCE EFFICIENCY

A key property of SignRAG is that retrieval is performed *offline*. Specifically, gallery search is conducted once before training or evaluation, and the retrieved results are stored in the data manifests. Therefore, no full-gallery search is performed during the deployed `model.generate` path.

The additional online inference cost mainly arises from the increased input length introduced by the retrieved textual contexts.

Table 11 reports the online generation efficiency on CSL-Daily. Compared with the Non-RAG baseline (`Pretrain→SFT`), SignRAG-SFT increases the average latency from 42.69 ms/sample to 61.98 ms/sample, corresponding to an additional 19.29 ms/sample. The throughput decreases from 23.42 to 16.13 samples/s, while the peak GPU memory increases from 23.28 to 29.03 GiB/GPU. This corresponds to a throughput reduction of approximately 31.13% and an additional memory consumption of 5.75 GiB/GPU. Despite this moderate overhead, SignRAG-SFT improves BLEU-4 on CSL-Daily from 29.72 to 31.22.

Table 11: **Online inference efficiency on CSL-Daily.** Both methods use Qwen3-8B and four NVIDIA RTX 4090D GPUs under the same inference configuration. Retrieval itself is performed offline and is therefore excluded from the deployed generation path.

Method	Latency (ms/sample)	Throughput (samples/s)	Peak GPU Mem. (GiB/GPU)
Non-RAG (`Pretrain→SFT`)	42.69	23.42	23.28
SignRAG-SFT	61.98	16.13	29.03

E.2 Offline Retrieval Efficiency on CSL-Daily

We next evaluate the efficiency of the offline retrieval stage. The retrieval gallery consists of the target-domain training samples, while each evaluation sample serves as a query. Table 12 reports the retrieval latency and throughput when progressively increasing the gallery size while keeping the same set of 256 queries.

As the gallery grows from 1K to 18.4K samples, the average retrieval latency increases from 3.49 to 23.88 ms/query. At the same time, retrieval throughput decreases from 286.67 to 42.02 queries/s. These results show that the retrieval cost increases with gallery size, but remains practical at the scale of CSL-Daily.

Table 12: **Offline retrieval efficiency on CSL-Daily with varying gallery sizes.** All settings use the same 256 queries.

Gallery Size	Latency (ms/query)	Throughput (queries/s)
1K	3.49	286.67
5K	7.10	140.80
10K	14.65	69.32
18.4K	23.88	42.02

For the complete CSL-Daily test set, retrieving 1,176 queries from the full gallery of 18,401 training samples requires 64.70 s in total, corresponding to 55.01 ms/query. The difference between this end-to-end retrieval latency and the 256-query micro-benchmark in Table 12 is caused by the complete retrieval pipeline, which additionally includes data loading, batching, result aggregation, and manifest construction. The peak active retrieval memory is 1.17 GiB per GPU, while the corresponding CPU memory usage is 10.42 GiB per rank.

Combining the one-time offline retrieval cost with SignRAG generation, the complete retrieval-and-generation evaluation on CSL-Daily requires 137.59 s for 1,176 test samples. This corresponds to approximately 117.00 ms/sample and 0.153 GPU-hours when accounting for all four GPUs. Since retrieval results can be pre-computed and reused, the deployed online inference latency remains the 61.98 ms/sample reported in Table 11.

E.3 Retrieval Scalability on OpenASL

To further evaluate scalability beyond CSL-Daily, we conduct the same retrieval analysis on OpenASL, whose training gallery contains approximately 96K samples. Table 13 reports the retrieval efficiency under progressively increasing gallery sizes.

With a gallery of 95,888 samples, the retrieval latency reaches 705.85 ms/query for the 256-query benchmark, corresponding to a throughput of 1.42 queries/s. For the complete OpenASL test set, retrieving 971 queries from the full 95,888-sample gallery requires 715.90 s in total, corresponding to 737.29 ms/query. These results demonstrate that token-level retrieval remains feasible at the approximately 100K-gallery scale, although its cost grows substantially with gallery size.

Table 13: **Offline retrieval efficiency at different gallery sizes on OpenASL.** The same 256 queries are used for all gallery-size experiments.

Gallery Size	Latency (ms/query)	Throughput (queries/s)
1,000	11.20	89.38
5,000	39.98	25.02
10,000	77.28	12.94
95,888	705.85	1.42

E.4 Training Cost

Table 14 summarizes the training cost of the complete SignRAG pipeline, with CSL-News serving as the pre-training dataset and CSL-Daily as the downstream fine-tuning dataset. The two-stage pre-training procedure requires 106.67 GPU-hours for Stage I and 277.50 GPU-hours for Stage II. Downstream SignRAG-SFT requires 24.04 GPU-hours, while SignRAG-RFT requires an additional 5.78 GPU-hours. Overall, the complete training pipeline requires 413.99 GPU-hours.

Table 14: **Training cost of SignRAG.**

Component	Cost
Pre-training Stage I	106.67 GPU-hours
Pre-training Stage II	277.50 GPU-hours
SignRAG-SFT	24.04 GPU-hours
SignRAG-RFT	5.78 GPU-hours
Total training compute	**413.99 GPU-hours**

E.5 Discussion on Million-Scale Retrieval

The above experiments demonstrate the practical feasibility of SignRAG on CSL-Daily and OpenASL, with gallery sizes of approximately 18K and 96K, respectively. However, token-level matching becomes increasingly expensive as the gallery grows. For substantially larger corpora such as BOBSL, which contains approximately 1.2 million sentences, we do not currently report direct retrieval results. At present, BOBSL is not available to us, and a reliable evaluation would additionally require data preparation, model pretraining, feature extraction, index construction, and retrieval-quality validation. Therefore, without direct experimental evidence, we avoid extrapolating the current measurements to the million-scale setting.

A potential direction for improving scalability is *coarse-to-fine retrieval*. For example, global video representations or domain information could first be used to shortlist a candidate pool through approximate nearest-neighbor search, after which the proposed fine-grained token-level matching would be applied only to the shortlisted candidates. Gallery sharding and batched retrieval could further reduce wall-clock latency.

However, coarse-to-fine retrieval introduces a non-trivial accuracy–efficiency trade-off. A coarse retrieval stage based on a single global representation may discard samples that would otherwise receive high scores under full token-level matching. Therefore, we do not claim that coarse-to-fine retrieval can fully preserve the accuracy of exhaustive SignRAG retrieval. A more comprehensive evaluation at the million-scale should jointly consider (i) retrieval latency, (ii) Top-K candidate preservation, (iii) retrieval quality, and (iv) downstream SLT performance under different candidate-pool sizes.

Finally, we emphasize that the primary contribution of SignRAG is not to claim unrestricted scalability to arbitrary million-scale open-domain corpora, but rather to extend retrieval-augmented SLT from settings that rely on gloss annotations to a gloss-free formulation and to demonstrate its effectiveness on multiple downstream benchmarks, including relatively large-scale datasets such as How2Sign and OpenASL. The OpenASL results in Table 13 directly characterize the computational behavior of the current implementation at the largest scale evaluated in this work.

For very large datasets, the limited vocabulary of sign language may also reduce the need for global Top-K retrieval. A fixed subset of the gallery may already provide useful retrieval candidates. However, this hypothesis still requires empirical validation. Million-scale retrieval therefore remains a limitation of the current work and an important direction for future study.

F More Cases

Due to space limitations in the main paper, we provide additional qualitative examples in Table 15. These cases offer a more comprehensive view of the translation performance of our method across diverse examples.

Source	Sentence
Reference	我想要找借口。 *(I want to find an excuse.)*
Retrieved	[1] 他随口编了一个迟到的借口。*(He casually made up an excuse for being late.)* [2] 为自己找借口的人，永远不会进步。*(People who make excuses for themselves will never improve.)* [3] 我不会用出差作为任何事情的借口。*(I would not use a business trip as an excuse for anything.)*
Geo-Sign	你找我有什么事吗? *(Why are you looking for me?)*
SignRAG	找借口。 *(Find an excuse.)*
Reference	封面设计是一种艺术设计。 *(Cover design is a form of artistic design.)*
Retrieved	[1] 一个好的封面设计会让你脱颖而出。*(A good cover design will make you stand out.)* [2] 我姐姐是从事封面设计工作。*(My sister works in cover design.)* [3] 我姐姐是从事封面设计工作。*(My sister works in cover design.)*
Geo-Sign	平面设计是一种艺术与技术的结合。 *(Graphic design is a combination of art and technology.)*
SignRAG	封面设计是一种艺术设计。 *(Cover design is a form of artistic design.)*
Reference	这个视频拍下了你作弊的行为。 *(This video recorded you cheating.)*
Retrieved	[1] 他对自己作弊的行为感到非常羞愧。*(He felt very ashamed of his cheating behavior.)* [2] 他对自己作弊的行为感到非常羞愧。*(He felt very ashamed of his cheating behavior.)* [3] 你的行动还不够迅速。*(Your actions are not quick enough.)*
Geo-Sign	他以借书为名，行偷盗之实。 *(He used borrowing books as a pretext for stealing.)*
SignRAG	你作弊的行为是可耻的。 *(Your cheating behavior is shameful.)*
Reference	今天水果很新鲜，买一个西瓜回家。 *(The fruit is very fresh today; buy a watermelon to take home.)*
Retrieved	[1] 天气热，买点水果回家给爸爸吃。*(The weather is hot; buy some fruit to take home for Dad.)* [2] 这些水果真新鲜，我们买西瓜还是苹果？*(These fruits are really fresh; should we buy a watermelon or apples?)* [3] 这些水果真新鲜，我们买西瓜还是苹果？*(These fruits are really fresh; should we buy a watermelon or apples?)*
Geo-Sign	水果今天很新鲜，买一串香蕉回家。 *(The fruit is very fresh today; buy a bunch of bananas to take home.)*
SignRAG	水果今天很新鲜，买一个西瓜回家。 *(The fruit is very fresh today; buy a watermelon to take home.)*

Table 15: Additional qualitative comparisons on CSL-Daily. Each example presents the reference translation, top-3 retrieved texts, and outputs of Geo-Sign and SignRAG. English translations are shown in parentheses below the original Chinese sentences.