Scaling Sign Language Translation

Biao Zhang **Garrett Tanzer** **Orhan Firat**
Google DeepMind
{biaojiaxing,gtanzer,orhanf}@google.com

Abstract

Sign language translation (SLT) addresses the problem of translating information from a sign language in video to a spoken language in text. Existing studies, while showing progress, are often limited to narrow domains and/or few sign languages and struggle with open-domain tasks. In this paper, we push forward the frontier of SLT by scaling pretraining data, model size, and number of translation directions. We perform large-scale SLT pretraining on different data including 1) noisy multilingual YouTube SLT data, 2) parallel text corpora, and 3) SLT data augmented by translating video captions to other languages with off-the-shelf machine translation models. We unify different pretraining tasks with task-specific prompts under the encoder-decoder architecture, and initialize the SLT model with pretrained (m/By)T5 models across model sizes. SLT pretraining results on How2Sign and FLEURS-ASL#0 (ASL to 42 spoken languages) demonstrate the significance of data/model scaling and cross-lingual cross-modal transfer, as well as the feasibility of zero-shot SLT. We finetune the pretrained SLT models on 5 downstream *open-domain* SLT benchmarks covering 5 sign languages. Experiments show substantial quality improvements over the vanilla baselines, surpassing the previous state-of-the-art (SOTA) by wide margins.

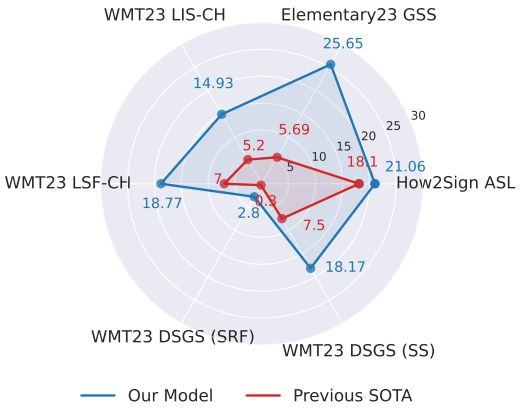


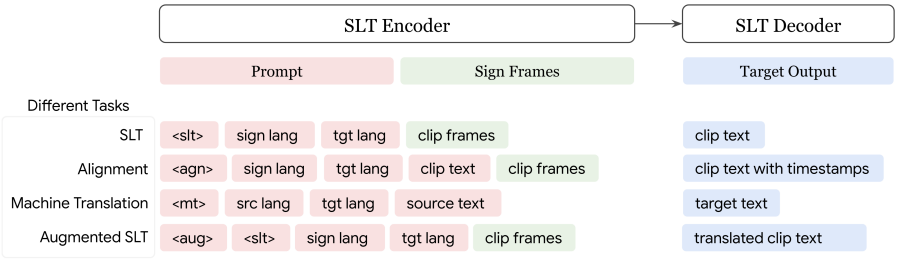
Figure 1: BLEU scores on different benchmarks: our model sets new SOTA results across benchmarks and sign languages. Note we didn't show BLEURT because not all previous studies report BLEURT.

1 Introduction

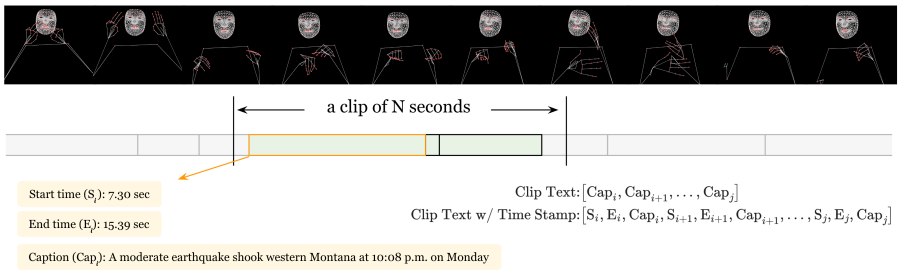
Scalable neural networks trained on large amount of unlabeled and/or weakly-labeled data from multiple modalities and multiple tasks have resulted in performance significantly exceeding that of single-task models trained on particular domains [30, 11, 33, 18]. Sign language translation (SLT),

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-3621



(a) Encoder-decoder based SLT model and different SLT pretraining tasks. We use red, green, and blue colors to indicate the input prompt, sign frames, and target output, respectively. "`sign lang`": sign language name; "`src lang/tgt lang`": source/target spoken language name; "<∗>": task-specific control tokens; "`source/target text`": source/target text for MT; "`clip frames (clip text)`": concatenation of sign frames (caption texts) corresponding to a video clip; "`translated clip text`": augmented data by off-the-shelf MT models; "`clip text with timestamps`": concatenation of caption texts and their start and end timestamps.



(b) Clip overview. Top: a sequence of skeletons for a sign language video where the used keypoints are annotated in red; Bottom: We pretrain SLT models on randomly sampled clips of N seconds from the video. Each segment in the plot represents a caption, and $[\text{Cap}_i, \ldots, \text{Cap}_j]$ (i.e., green segments) denotes captions fully covered by the clip. "S_i/E_i": the start/end time stamp for caption Cap_i.

Figure 2: Illustration of model architecture and pretraining task for SLT. We perform large-scale pretraining and adopt multi-task learning at clip level (multiple captions) to better leverage the supervised knowledge.

as a video-to-text translation task[1], features significant cross-modality challenges in video understanding and text generation. While extra forms of supervision such as glosses have been helpful in bridging the modality gap [4], they are nonstandardized/incomplete systems available only for small datasets [12]. Researchers have instead turned to more scalable approaches such as adapting pretrained vision and text models [7, 37, 50] and jointly modeling with machine translation [MT, 58]. Despite encouraging progress, these studies were performed at small scale with success on narrowed domains and on few sign languages. In open-domain SLT settings, unfortunately, they have shown limited effectiveness [28].

In this paper, we aim to improve *open-domain* SLT for multiple sign languages by means of large-scale SLT pretraining with more data, larger models and more languages. Inspired by the finding that jointly training SLT models with MT data enables positive knowledge transfer to SLT [58], we explore the following pretraining tasks and data: *web-crawled multilingual SLT*, *multilingual MT*, and *augmented SLT*. Although high-quality SLT training data are scarce, weakly-labeled SLT data covering diverse topics and signers are readily available from platforms like YouTube. Prior studies have demonstrated the feasibility of collecting massive YouTube SLT data and its effectiveness on improving SLT [46, 44, 43], and we follow this effort in a multilingual setup. Different from SLT, text-based MT datasets are massive and resource-rich across hundreds of spoken languages [3, 10]. We explore a subset of MADLAD-400 [23] including up to 41 spoken languages for the pretraining. In addition, we construct synthetic multiway SLT data by translating video captions with an off-the-shelf MT model, which allows us to strengthen direct SLT across more translation directions. We investigate different ways of mixing these data to exploit weakly supervised SLT knowledge as well as cross-task cross-modality transfer at scale.

[1]While spoken language can be conveyed through either text or speech, this study focuses on text.

https://doi.org/10.52202/079017-3621

As in Figure 2, we extend the unified encoder-decoder SLT framework from [46, 42, 44, 43] with extra tasks and modalities similar to [58], across different pretrained model families (T5, mT5 and ByT5) and different model sizes. We distinguish different tasks by carefully designed input prompts that contain task-specific control tokens. This affords us high flexibility in choosing what tasks and languages to incorporate into the pretraining, easing ablations and the scaling. We then finetune the pretrained SLT models on downstream SLT benchmarks to refine the learned SLT knowledge.

We evaluate the effect of scaling on 6 open-domain SLT benchmarks across 5 sign languages. FLEURS-ASL#0 [42], built on FLORES-200 [10], gives us a testbed to analyze multiway American Sign Language (ASL)-to-X SLT (we examine English and 41 other target languages), while the other benchmarks are for a single language pair. While pretraining results show the acquired general SLT capability, we also report finetuning results following [46]. Our main findings are below:

- Adding more pretraining data, either machine translation or sign language translation data, is a promising way to improve SLT, yielding quality gains of varying degrees.
- Zero-shot ASL-to-X translation for language pairs not seen during pretraining is achievable by jointly training on ASL-to-En SLT data and En-to-X MT data.
- Augmenting SLT data by translating target captions to other languages with off-the-shelf MT models substantially improves the translation.
- Using larger models is not always helpful: ByT5 Base (582M) often outperforms XL (3.74B), but model scaling does benefit SLT when modeling capacity becomes a bottleneck (e.g., when more languages and data are used).
- Learned metrics (e.g., BLEURT) show higher correlation between pretrained and finetuned SLT scores than classical metrics (e.g., BLEU or ChrF).

Putting everything together, our model achieves new state-of-the-art results across the benchmarks as shown in Figure 1, demonstrating the significance of scaling SLT.

2 Sign Language Translation

2.1 Modeling

We build on a line of work using T5 model families for SLT [46, 42, 44, 43], which build upon earlier SLT work [4, 61, 58] using the encoder-decoder architecture [41, 47]. Figure 2a shows the overall structure. The encoder takes as input the concatenation of a prompt instructing the task and a sequence of sign language video frames; the decoder predicts the text output in a target spoken language one token at a time. We adopt the family of pretrained (m/By)T5 models [35, 53] as the backbone and adapt them to SLT via large-scale SLT pretraining followed by downstream finetuning, i.e. **(m/By)T5 initialization** $\rightarrow$ **SLT pretraining** $\rightarrow$ **SLT finetuning**.

We rely on web-crawled YouTube SLT data for SLT pretraining, which provide high coverage on domains and signers albeit at lower quality. Although recent debates value data quality over data quantity in pretraining [21, 24, 29], we argue that they were established on the availability of massive high-quality training data, which doesn't hold for SLT yet. We expect that the pretraining could capture the (weakly) supervised SLT knowledge from the crawled data as in previous studies [46].

As shown in Figure 2b, we adopt the clip-level training following [42] that randomly samples a clip of N seconds from the sign video and then predicts various types of in-clip information (such as caption texts and their start and end timestamps) based on the frames of the entire clip. Detailed tasks are listed in Figure 2a, which are all formulated as sequence-to-sequence tasks. They are distinguished by prompts with different control tokens and are trained with the standard maximum likelihood objective. For the **baseline**, we consider the following two tasks: *SLT* and *alignment*, and train it by mixing these two tasks with a pre-specified mix ratio.

SLT This is the core task that directly models the translation from clip frames to the clip text in a target language. It is indispensable for the model to acquire the translation capability.

Alignment It is an auxiliary task for SLT, learning to align the input clip with its captions. We train the model to infer the start and end time stamp for each in-clip caption. Apart from regularization, this task could improve the model's understanding of sign language [42].

 https://doi.org/10.52202/079017-3621

2.2 Scaling Model Size, Number of Languages and Pretraining Data Size

Model Scaling Scaling model size increases modeling capacity, which has been widely proven effective in improving the task performance [30, 18, 19]. We study whether and how increasing model size affects the SLT performance and compare (By/m)T5 models for SLT at different scales.

Language Scaling While most SLT works focus on a few sign and spoken languages, we expand our study to massive languages, covering up to 80 sign/spoken languages during pretraining, and 5 sign language and 42 spoken languages at evaluation. We are interested in whether a single SLT model could support multiple sign/spoken languages with non-trivial performance, and whether knowledge transfer could improve SLT on low-resource languages [57, 44].

Data Scaling Data scarcity is the main bottleneck hindering the development of SLT. To address this issue, we investigate the following three types of data for the pretraining:

SLT We crawl multilingual YouTube SLT data following the recipe [44] except that we didn't perform human annotation and filtering. This allows us to significantly scale up the SLT data by 3∼6 times, reaching ∼6,600 hours in total, albeit at much lower quality.

Machine Translation Unlike SLT, MT is a text-to-text translation task with rich parallel resources, particularly for high-resource languages [2]. We explore adding multilingual MT data into the pretraining and mark this task with control token "`<mt>`" [58].

Augmented SLT SLT data are often one-to-one translation data, where each sign language only has translation in one spoken language. This makes the translation of a sign language to other spoken languages difficult. We thus augment SLT data to one-to-many by translating the target text to other spoken languages via off-the-shelf MT models. As in Figure 2a, we use "`<aug>`" to separate genuine SLT data from the augmented one [6].

3 Setup

MT Pretraining Data We use the parallel sentence-level portion of MADLAD-400 [23] as the MT pretraining data. We extract a subset of MADLAD-400 for experiments, including 41 languages (apart from English (En)) covering diverse language families and scripts, and explore the impact of En→Xx and Xx→En MT data on SLT in experiments. We create two settings for the pretraining:

- **MT-Small**: A high/medium-resource subset including 11 languages `es, de, fr, it, cs, pl, ru, zh, ar, ja, hi`.
- **MT-Large**: This set includes all 41 languages. Apart from MT-Small, it has `nl, pt, sv, hu, da, fi, el, sk, no, bg, lt, lv, sl, et, ko, hr, is, sr, tr, vi, id, he, th, ms, uk, ca, ta, fil, ne, cy`.

Table 7 shows the statistics for each language. Unless otherwise specified, we balance the MT data distribution over languages during training by temperature sampling with a rate of 5 [2].

SLT Pretraining Data We experiment with noisy captioned sign language videos from YouTube. This is the full set of videos pre-manual filtering in [44]. Estimated statistics for each sign language are summarized in Table 7. We also have two settings for this data:

- **YT-ASL**: ∼2,800 hours of noisy captioned ASL videos; a superset of YouTube-ASL [46] (modulo video churn) and the same dataset used by [43].
- **YT-Full**: ∼6,600 hours of noisy captioned multilingual sign language videos; a superset of [44].

During training, we mix the SLT data for all languages in proportion to their duration. We further augment these data with other spoken languages via MADLAD-MT-3B [23]. For ASL SLT data, we translate the English captions to 41 spoken languages listed in MT-Large, which makes YT-ASL 43-way multilingual SLT, namely **Aug-YT-ASL**; for other SLT data, we translate the target text into English, resulting in 3-way multilingual SLT.[2] We refer to the augmented SLT data for all sign

[2]Note translations were performed per caption, which may lack coherence when compiled into a document.

Task	Sign Lang	Target Lang	#Train	#Dev	#Test
How2Sign	ASL	En	183,097	10,277	13,890
Elementary23	GSS	El	35,970	512	512
	LIS-CH	It	1,901	100	250
WMT23	LSF-CH	Fr	5,560	100	250
	DSGS	De	310,840	420 (WMT22)	250/246 (SS/SRF split)
FLEURS-ASL#0	ASL	200 Flores Langs	-	-	353

Table 1: Summary of downstream SLT benchmarks. "#Train/#Dev/#Test": the number of examples in the train, dev and test split. Note the sign language video and the target text in these benchmarks are often pre-segmented and aligned at sentence level. "DGS/ASL/GSS": German/American/Greek Sign Language; "En/De/Fr/It": English/German/French/Italian; "LIS-CH": Italian Sign Language of Switzerland; "LSF-CH": French Sign Language of Switzerland; "DSGS": Swiss German Sign Language.

languages as **Aug-YT-Full**. Similar to MT-Small and MT-Large, we reorganize the augmented data to **Aug-YT-ASL-Small/Aug-YT-Full-Small** and **Aug-YT-ASL-Large/Aug-YT-Full-Large**.

SLT Pretraining Mixture We ablate across several SLT pretraining mixtures.

- **Baseline**: Caption alignment and SLT tasks. We use the task weights from [42], including 4% for alignment.
- **Baseline + MT**: We mix MT data into Baseline with a sampling probability of p_{MT}.
- **Baseline + Augmented SLT**: We replace the Baseline SLT data with the augmented SLT data and uniformly sample the target language for each example at each step.
- **Baseline + MT + Augmented SLT**: *Baseline + MT* but with augmented target languages, as above.

Downstream Benchmarks, Evaluation and Model Setting We thoroughly evaluate the translation performance on a range of *open-domain* SLT benchmarks, including How2Sign [14], Elementary23 [48][3], WMT23 [28] and FLEURS-ASL#0 (signer id #0) [42]. Detailed information for each benchmark is given in Table 1. Overall, the evaluation covers 5 source sign languages and 42 target spoken languages.[4]

We report translation results for **Pretraining** and **Finetuning**. During inference, we use beam search with a beam size of 5. We evaluate translation with detokenized BLEU [31] and ChrF [32], as well as neural metric, BLEURT [34]. We use BLEURT as the main metric [17]. We initialize our SLT model with three T5 model families: T5 [35], mT5 [52] and ByT5 [53], at three different sizes: Base, Large and XL. We optimize models with Adafactor [39], and set the maximum text input, landmark input, and text output length to 512. More setup details are given in Appendix A.1.

4 Experiments

4.1 SLT Pretraining Results

Model scaling doesn't improve SLT consistently: Base often outperforms Large/XL. Table 2 also shows that scaling up model size rarely results in consistent quality improvements. Different from findings on text-only tasks [35, 53], Base surpasses Large and XL in most cases, where Large often converges the slowest and performs the worst. Model scaling alone doesn't significantly reduce the video-text modality gap, although better optimization and checkpoint selection could help. XL performs relatively comparable to Base. When MT data is mixed in and modeling capacity becomes the bottleneck, the value of model scaling by XL emerges as shown in Figure 8 and Table 5.

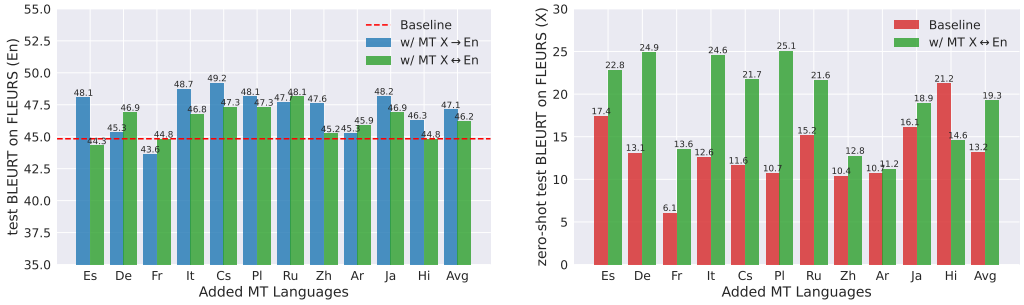
[3]While not as restricted as specific domains like "weather forecasts", the scope of topics in Elementary23 remains somewhat focused.

[4]We acknowledge that there are other SLT benchmarks available in academia. We didn't include them in our experiments due to their licensing restrictions and/or domain limitations.

https://doi.org/10.52202/079017-3621

SLT Data	Model	How2Sign			FLEURS-ASL#0 (En)		
		Base	Large	XL	Base	Large	XL
YT-ASL	T5	**29.54**	27.95	22.96	**32.8**	4.18	32.41
	mT5	**34.94**	8.46	23.7	35.59	**43.53**	23.09
	ByT5	**30.36**	23.51	29.2	**44.84**	28.47	41.65
YT-Full	T5	**31.64**	25.45	8.57	**42.86**	37.55	30.02
	mT5	**31.46**	19.37	24.46	**38.03**	24.56	33.16
	ByT5	**37.13**	22.61	29.59	52.48	43.01	**52.71**

Table 2: Pretraining performance (BLEURT ↑) for different sized (By/m)T5 models when pretrained on YT-ASL and YT-Full. Results are reported on the test set of How2Sign and FLEURS-ASL#0 (→En, i.e. English as the target). Best results for each model family are highlighted in bold.



(a) BLEURT scores for FLEURS-ASL#0 (→En). (b) Zero-shot BLEURT for FLEURS-ASL#0 (→X).

Figure 3: Pretraining performance for *Baseline + MT* when varying MT languages. We show BLEURT↑ results on FLEURS-ASL#0, and set $p_{mt} = 0.5$. Note MT languages are added separately instead of jointly. Results are for ByT5 Base. "X→En": MT data for translation into English; "X↔En": MT data for both translation directions; "Avg": average performance over languages. MT languages are arranged in descending order from left to right based on their training data quantity.

Backbone affects SLT substantially; ByT5 generally performs the best. While several previous studies selected T5 [46, 28] or mT5 [44] as the SLT backbone, we observe in Table 2 that ByT5-based SLT outperforms its T5 counterpart in most settings, confirming the results of [43] at scale. Given that larger models do not consistently perform better, it seems less likely that ByT5's superiority comes from its encoder-heavy parameter allocation, and more likely that it is due to its spelling capabilities and reduced input length gap between byte text sequences and video frame sequences. Unless otherwise stated, we use ByT5 Base for the following experiments.

Scaling SLT data generally improves quality significantly. Adding more SLT data, i.e. from YT-ASL to YT-Full, largely improves the translation quality in most settings. For ByT5-based SLT particularly, the gain reaches ∼7 BLEURT on How2Sign and ∼11 BLEURT on FLEURS-ASL#0 (En) for Base and XL, respectively. We conjecture that adding more (multilingual) SLT data helps reduce the modality gap (especially with skeletons, which lack pretrained representations) and enable cross-lingual knowledge transfer [2, 57, 55, 44].

Mixing MT and SLT data yields positive knowledge transfer to SLT. We next explore whether and how the addition of MT data benefits SLT, starting with YT-ASL and bilingual MT data with $p_{mt} = 0.5$. Figures 3a and 5 show that adding bilingual translation data improves SLT performance generally, confirming the findings of SLTUNet [58]—that jointly training with MT enables positive knowledge transfer—at scale. The quality gains vary greatly across languages, which show little correlation with language family or training data scale. For example, adding a large amount of Fr→En data (∼243M sentence pairs) helps little (or even hurts) on FLEURS-ASL#0 (En), while adding a small amount of Ja→En data (∼5M sentence pairs) gives a gain of at least 3 BLEURT on How2Sign and FLEURS-ASL#0 (En).

Translation direction of MT data affects transfer to SLT. There are three ways to leverage MT data for SLT: 1) X→En, 2) En→X, and 3) both. We compare 1) and 3) in Figures 3a and 5 for ASL-to-En SLT. The translation direction of MT data influences SLT performance greatly and varies

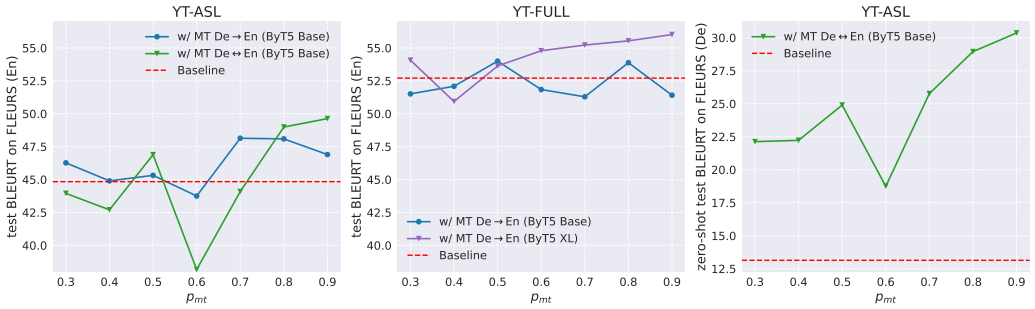


Figure 4: Pretraining performance for *Baseline + MT* when changing the mixing ratio of MT data p_{mt} on FLEURS-ASL#0 (En and De) test set. We show BLEURT↑ results as we vary p_{mt} from 0.3 to 0.9.

across languages. On average, X→En benefits ASL-to-En SLT more than X↔En: +0.08 and +0.9 BLEURT on How2Sign and FLEURS-ASL#0 (EN), respectively. We speculate that including translation into X uses model capacity, which, while enabling zero-shot ASL-to-X SLT as discussed below, results in slightly worse ASL-to-En performance. This suggests that MT data with the same target language as SLT is most effective for transfer. Table 3 shows further support where En→X surpasses X→En on multilingual SLT.

We can achieve zero-shot bilingual ASL-to-X SLT via ASL-to-En SLT + En↔X MT, albeit at poor quality. If knowledge can be transferred from MT to SLT, one straightforward question is whether we can achieve zero-shot SLT by jointly training with MT. We do so by training on ASL-to-En SLT + En↔X MT data and examining zero-shot ASL-to-X SLT on FLEURS-ASL#0 (X). Figure 3b shows that this works effectively. On Pl and It, we observe quality gains over 12 BLEURT; on average, adding MT data improves zero-shot SLT by ∼6 BLEURT. Nevertheless, the overall zero-shot SLT performance is middling, and the gains are unstable across languages, e.g. performance degrades for ASL-to-Hi SLT with joint MT training. Similar findings were also observed in multilingual MT and speech translation [57, 13]. Deeper analysis in Appendix A.2 reveals that zero-shot SLT also suffers from the off-target translation problem [57], i.e. translating into a wrong target language; adding MT data can alleviate it, mostly for high-resource languages.

SLT Data	Dir	BLEURT	
		Small	Large
Baseline + YT-ASL		15.85	17.21
Baseline + YT-Full		24.36	23.16
Baseline + MT-Small			
YT-ASL	En→X	23.51	21.25
	X→En	17.44	19.72
	En↔X	23.84	19.19
YT-Full	En→X	27.29	23.15
	X→En	22.48	22.47
	En↔X	26.33	21.48
Baseline + MT-Large			
YT-ASL	En↔X	24.69	26.60
YT-Full	En↔X	**29.52**	**30.69**

Table 3: Pretraining performance for *Baseline + MT* with $p_{mt} = 0.9$ when scaling up languages and data. We show averaged BLEURT↑ results on FLEURS-ASL#0. Results are for ByT5 Base. "Dir": translation direction of MT data; "Small-/Large": average results over the target languages included in MT-Small/MT-Large on FLEURS-ASL#0.

Using a higher sampling ratio for the MT data, i.e. larger p_{mt}, often improves SLT. We start with $p_{mt} = 0.5$, i.e., sampling equal amount of SLT and MT data, in the above experiments following intuition. However, the proportion of different types of data often has non-negligible influence in multilingual modeling [2, 9]. We next explore its impact on SLT and use MT En-De for illustration. Figure 4 and 6 shows that $p_{mt} = 0.5$ is sub-optimal and sampling more MT data improves SLT in most settings, regardless of using ByT5 Base or XL, YT-ASL or YT-Full, MT De→En or De↔En, and How2Sign or FLEURS-ASL#0 (En/De). In addition, increasing the proportion of MT data also improves zero-shot ASL-to-De SLT. Note another benefit of using more MT data is to accelerate training, as loading SLT data is much slower than loading text-only MT data. We use $p_{mt} = 0.9$ by default in the following experiments.

Multilingual MT improves multilingual (zero-shot) SLT. The above experiments mainly analyze SLT with bilingual MT. We next investigate how multilingual MT affects multilingual (zero-shot) SLT, particularly the use of MT-Small and MT-Large. We report results for ASL-to-*Small* and ASL-to-*Large* SLT on FLEURS-ASL#0 where Small and Large denote the target languages covered by

https://doi.org/10.52202/079017-3621

MT-Small and MT-Large, respectively. Note all SLT directions are zero-shot except the translation to English.

Table 3 summarizes the average performance. Using multilingual X→En MT data results in unstable ASL-to-X SLT performance, which even hurts SLT on YT-Full. In contrast, multilingual En→X and En↔X MT data are both very helpful to SLT, where the former often outperforms the latter. By default, we still use En↔X MT data in the following experiments so as to fully leverage the knowledge in MT data during pretraining.

Figures 7a and 7b further show the language breakdown results. Adding multilingual MT significantly improves ASL-to-En SLT when using YT-ASL alone, while the gain almost disappears when using larger-scale SLT data, YT-Full. Again, we note that the overall zero-shot translation quality is poor – the best average BLEURT on Small and Large is 29.52 and 30.69, respectively. Achieving significant ASL-to-X SLT requires techniques beyond naive SLT and MT data mixing.

Data augmentation and large-capacity modeling are promising methods for multilingual SLT. In MT, a common solution to improve zero-shot quality is to construct pseudo translation data for zero-shot directions [2, 15, 56, 16]. We examine this practice for SLT. We adopt publicly pretrained MT models to generate data for more target languages for the YouTube SLT data (i.e., Augmented SLT). Results in Table 4 demonstrate the effectiveness of Augmented SLT, which significantly improves the best performance for ByT5 Base-based SLT to 36.01 and 39.85 average BLEURT on Small and Large with a gain of 6.49 (29.52→36.01) and 9.16 (30.69→39.85), respectively. Note there are 42 languages in Large. ByT5 Base may be insufficient in accommodating translation for such amount of languages. Increasing the modeling capacity to XL yields another gain of 9.11 (36.01→45.12) and 8.2 (39.85→48.05) average BLEURT on Small and Large, respectively. On YT-ASL, Augmented SLT and ByT5 XL also lead to substantial quality improvements by 13.84 (24.69→38.53)/15.96 (26.60→42.56) average BLEURT on Small/Large. The final performance even surpasses the cascading baseline, i.e. ASL-to-En SLT chained with En-to-X MT, under both YT-ASL and YT-Full. Figures 8a and 8b also show the quality improvements across languages resulted from data augmentation and ByT5 XL.

Setting	BLEURT	
	Small	Large
Baseline + YT-ASL	15.85	17.21
+ Aug-YT-ASL-Small	31.14	19.74
+ MT-Small	30.51	19.70
+ Aug-YT-ASL&MT-Large	25.83	33.71
+ ByT5 XL	**38.53**	**42.56**
+ MT-3B Cascading	34.82	37.82
Baseline + YT-Full	24.36	23.16
+ Aug-YT-Full-Small	38.53	29.49
+ MT-Small	36.84	25.67
+ Aug-YT-Full&MT-Large	36.01	39.85
+ ByT5 XL	**45.12**	**48.05**
+ MT-3B Cascading	43.54	46.32
+ ByT5 XL	44.82	47.52

Table 4: Pretraining performance (averaged BLEURT↑) for *Baseline + Augmented SLT + MT* with $p_{mt} = 0.9$ on FLEURS-ASL#0 test set. MT data are multilingual in both directions. Baseline is for ByT5 Base; "MT-3B": MADLAD-MT-3B, the model used for SLT augmentation; "Cascading": translating FLEURS-ASL#0 to English and then performing MT to other target languages.

4.2 SLT Finetuning Results

We report the finetuning performance measured by BLEURT in Table 5. We also include the BLEU and ChrF results as well as the corresponding pretraining results in Appendix (Tables 8 and 9).

Finetuning on downstream benchmarks substantially improves SLT performance. Table 5 shows that finetuning the pretrained SLT models yields substantial quality gains across benchmarks and settings. This is because the potential of pretrained models is not fully elicited by direct evaluation due to video recording, domain and (clip-based) pretraining vs. (segment-based) inference mismatches, and finetuning largely mitigates these gaps. For example, pretraining with external augmented SLT and MT data results in even worse pretraining performance ((6)→(7)) in Table 9. After finetuning, nevertheless, model (7) significantly surpasses model (6) by 5.12 BLEURT on average.

Adding multilingual SLT data (YT-Full) into the pretraining greatly improves the performance from 14.26 (model (5)) to 32.48 BLEURT (model (8)) in Table 9. However, the quality gain after finetuning for YT-ASL based models is often higher than their YT-Full counterparts, where the largest gain reaches ∼28 BLEURT for model (5). We argue that pretraining on YT-ASL mainly teaches

ID	Model	H2S	E23	WMT23				Avg
				LIS-CH	LSF-CH	SRF	SS	
0	Prevous SOTA	50.80	-	25.20	18.80	24.60	37.70	-
1	ByT5 Base	34.00	22.14	22.77	7.74	15.41	26.88	21.49
2	1 + Baseline + YT-ASL	51.74	37.79	24.24	15.43	21.82	35.59	31.10
3	2 + MT-Small ($p_{mt} = 0.9$)	52.62	45.98	33.10	24.58	23.33	45.45	37.51
4	3 + Aug-YT-ASL-Small	53.36	49.34	38.61	28.70	25.87	49.61	40.91
5	4 + Aug-YT-ASL&MT-Large + ByT5 XL	54.28	54.16	38.93	27.29	28.42	51.73	42.47
6	2 + YT-Full	53.51	49.48	42.11	31.16	21.15	44.28	40.28
7	6 + Aug-YT-ASL&MT-Small	53.70	53.13	45.09	37.69	30.31	52.45	45.40
8	7 + Aug-YT-ASL&MT-Large + ByT5 XL	**55.69**	**56.94**	51.94	**41.14**	**33.94**	57.96	**49.60**
9	8 + Multilingual SLT Tuning	53.47	55.57	**54.54**	39.26	29.33	**58.08**	48.38

Table 5: Finetuning performance (BLEURT↑) on downstream SLT benchmarks. "H2S/E23": How2Sign/Elementary23. "SRF/SS": WMT23 DSGS SRF/SS test split. "Avg": averaged performance over all benchmarks. MT data are added in both translation directions. Previous SOTA: How2Sign [43], Elementary23 [48] and WMT23 SRF [28], WMT23 LIS-CH, LSF-CH, SS [44]. *All models are finetuned on each SLT benchmark separately except (9).*

understanding of ASL, so pretrained performance on other sign languages is poor, but finetuning can quickly adapt the learned representations to other sign languages.

Note we also finetuned the vanilla ByT5 model without SLT pretraining for reference, which achieves 7.68 and 3.10 BLEU on Elementary23 and WMT23 DSGS SS, respectively. Despite their inferiority, these results already surpass the previous SOTA, further showing the potential of ByT5.

A model's pretraining performance may be misleading when estimating its downstream finetuning performance, depending on the evaluation metric. Intuitively, a model with better pretrained results should result in better finetuned results. The Spearman's correlation results in Table 6 confirm this intuition, where the correlation scores are positive across metrics. However, BLEU and ChrF have a correlation score of 0.347 and 0.186, respectively, which are very moderate. The correlation for ChrF is even not significant, which may be caused by the use of BLEU as the model selection metric. In contrast, the correlation of BLEURT reaches 0.578 and is significant at $p < 0.01$.

	BLEU	ChrF	BLEURT
Spearman's ρ	$0.347^{\dagger}$	0.186	$\mathbf{0.578}^{\ddagger}$

Table 6: Spearman correlation between direct (i.e. pretraining) and finetuning SLT results under different metrics based on Tables 5 and 9. $^{\dagger}/^{\ddagger}$: significant at $p < 0.05/0.01$.

Model, data and language scaling together leads to new state-of-the-art results. Diving deeper into Table 5, we see clear improvements brought by scaling model size, data, and/or languages for SLT. Adding YT-ASL SLT data into the pretraining yields ∼10 average BLEURT improvement ((1)→(2)). Jointly training SLT with MT data produces another gain of ∼6 BLEURT ((2)→(3)). Data augmentation adds an improvement of ∼3 BLEURT ((3)→(4)), which matches the quality achieved by adding large amount of extra multilingual SLT data to the baseline, i.e. (4) 40.91 vs. (6) 40.28. By further increasing the amount of MT and augmented SLT data as well as the ByT5 model size, we reach an average BLEU, ChrF and BLEURT of 16.90, 39.49, and 49.60, respectively (model (8)). These results also outperform previous best results, establishing the new SOTA.

Multilingual finetuning improves multilingual SLT with encouraging performance, although it still underperforms bilingual finetuning on average. We next study multilingual finetuning on the direct mix of different SLT benchmarks. Table 5 ((8)→(9)) shows that multilingual SLT outperforms previous SOTA on almost all benchmarks, but underperforms its bilingual counterpart by 1.22 BLEURT on average. How to balance modeling capacity among different languages in a joint model and avoid cross-lingual/modality interference is a well known issue in multilingual modeling [2, 57, 49], and multilingual SLT also suffers [55], which we leave to future. Still, multilingual SLT facilitates transfer to LIS-CH, leading to a substantial gain of 2.6 BLEURT ((8)→(9)).

5 Related Work

The main bottleneck of SLT is data scarcity. Early studies address this issue by developing more data efficient neural architectures and/or training algorithms. Camgoz et al. [4] pioneered the study with

https://doi.org/10.52202/079017-3621

encoder-decoder based recurrent models for SLT, which was quickly replaced by Transformer and multi-task learning with CTC regularization [5]. Zhou et al. [61] developed spatial-temporal architecture to model the collaboration of different visual cues. Another way is to transfer the knowledge from pretrained models, augmentations, and other tasks. Chen et al. [7, 8] proposed to leverage pretrained visual encoders and MT models to improve SLT, while Zhang et al. [58] explored transferring translation knowledge from MT data directly. Zhou et al. [60] employed back-translation to generate pseudo SLT training data. Ye et al. [54] augmented the training data by the mix-up algorithm. Yet another way to address data scarcity is to make data less scarce. Shi et al. [40], Uthus et al. [46], and Tanzer and Zhang [44] collected large-scale SLT data from YouTube and improved data quality via manual filtering; Albanie et al. [1] developed a British Sign Language translation corpus based on BBC broadcasts instead. Tanzer [43] scaled up ASL data by eschewing manual filtering and tolerating misaligned or irrelvant data. We follow and scale to noisy multilingual sign language data, MT data, and augmented paralel data.

Despite the aforementioned advancements, many studies still heavily depend on *sign glosses*. As a bridge between sign video and target text, sign glosses ease learning, but are expensive to annotate, not always available, nonstandardized, and cannot cope with sign language grammar in generality [12]. Recent research therefore turns to gloss-free SLT, which often underperforms gloss-based counterparts [25, 59, 50] and performs poorly in open-domain settings [38, 51, 28]. We substantially improve gloss-free SLT performance across benchmarks through scaling. In this regard, our work is closely related to SSVP-SLT [37] but with different focuses. SSVP-SLT improves SLT by pretraining a neural sign encoder through large-scale self-supervised learning. By contrast, we adopt static landmarks to represent sign frames and improve the translation by transferring knowledge from other languages and tasks. The methods used in our study are orthogonal to SSVP-SLT. In addition, our work also falls into the category of improving multilingual SLT [55, 20]. We didn't evaluate our models on these multilingual benchmarks though as they are either unavailable at the time of paper writing or unusable due to licensing issues.

6 Conclusion, Limitations, and Future Work

We presented a systematic study of data, model and language scaling for SLT via large-scale SLT pretraining. In general, scaling substantially improves SLT. We observe positive knowledge transfer from other sign language data and from machine translation data. By joint SLT and MT training, we show the feasibility of achieving zero-shot SLT. Data augmentation expanding SLT data to more spoken languages via off-the-shelf MT models significantly improves multilingual SLT. Putting everything together, finetuning our pretrained SLT models leads to new state-of-the-art SLT results across 5 benchmarks covering 5 sign languages (but still far from usable quality).

Although our models have nominally been pretrained on a massive number of sign languages (up to 80), we lack comprehensive and reliable multilingual benchmarks to fully understand their abilities and limitations. In addition, our models are limited to encoder-decoder based (m/By)T5 models, and SLT pretraining requires many computational resources, increasing the difficulty of reproduction.

In the future, we expect that continuing to scale sign language data, number of sign languages, vision pretraining/multimodality, etc. will reap further gains. As suggested by [42], it will be important to evaluate these growing capabilities on multilingual, standardized, open-domain SLT benchmarks.

Ethics Statement

We preprocess all sign videos with simplified landmarks as a form of anonymization and privacy protection. While the pretraining SLT data is larger scale than prior work, it may still suffer from demographic biases. Even if demographics were represented in proportion to the real world, and even with simplified landmarks, the resulting SLT models may not perform equally across groups and should be evaluated for fairness before real-world deployment. Our study mainly aims to understand the impact of scaling on SLT, and while we significantly improve translation quality, it is still far from usable for real-world applications. For many such applications, the other half of sign language translation—sign language generation—is also essential, whereas we focus only on sign language understanding in this work. Advancing both of these is critical to ensure that Deaf/Hard of Hearing signers get equal access to technology and the information that comes through it.

Acknowledgements

We thank the reviewers for their insightful comments. We thank Ankush Garg for valuable feedback on this work, Chris Dyer for constructive comments that greatly improve the quality of this paper, Sam Sepah and Google Translate team for supporting this research. We also thank the T5X team [36] for infrastructure support.

References

[1] Samuel Albanie, Gül Varol, Liliane Momeni, Hannah Bull, Triantafyllos Afouras, Himel Chowdhury, Neil Fox, Bencie Woll, Rob Cooper, Andrew McParland, et al. Bbc-oxford british sign language dataset. arXiv preprint arXiv:2111.03635, 2021.

[2] Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. Massively multilingual neural machine translation in the wild: Findings and challenges, 2019.

[3] Ankur Bapna, Isaac Caswell, Julia Kreutzer, Orhan Firat, Daan van Esch, Aditya Siddhant, Mengmeng Niu, Pallavi Baljekar, Xavier Garcia, Wolfgang Macherey, et al. Building machine translation systems for the next thousand languages. arXiv preprint arXiv:2205.03983, 2022.

[4] 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, pages 7784–7793, 2018.

[5] 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, pages 10023–10033, 2020.

[6] Isaac Caswell, Ciprian Chelba, and David Grangier. Tagged back-translation. In Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, André Martins, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Matt Post, Marco Turchi, and Karin Verspoor, editors, Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers), pages 53–63, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5206. URL https://aclanthology.org/W19-5206.

[7] 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, pages 5120–5130, 2022.

[8] 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, 2022.

[9] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. arXiv preprint arXiv:1911.02116, 2019.

[10] Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. No language left behind: Scaling human-centered machine translation. arXiv preprint arXiv:2207.04672, 2022.

[11] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In International Conference on Machine Learning, pages 7480–7512. PMLR, 2023.

 https://doi.org/10.52202/079017-3621

[12] Aashaka Desai, Maartje De Meulder, Julie A. Hochgesang, Annemarie Kocab, and Alex X. Lu. Systemic biases in sign language ai research: A deaf-led call to reevaluate research agendas, 2024.

[13] Tu Anh Dinh. Zero-shot speech translation. arXiv preprint arXiv:2107.06010, 2021.

[14] Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. How2Sign: A Large-scale Multimodal Dataset for Continuous American Sign Language. In Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[15] Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. Beyond english-centric multilingual machine translation. Journal of Machine Learning Research, 22(107): 1–48, 2021.

[16] Markus Freitag and Orhan Firat. Complete multilingual neural machine translation. arXiv preprint arXiv:2010.10239, 2020.

[17] Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In Philipp Koehn, Loïc Barrault, Ondřej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Alexander Fraser, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Tom Kocmi, André Martins, Makoto Morishita, Christof Monz, Masaaki Nagata, Toshiaki Nakazawa, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Marco Turchi, and Marcos Zampieri, editors, Proceedings of the Seventh Conference on Machine Translation (WMT), pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.wmt-1.2.

[18] Team Gemini, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805, 2023.

[19] Team Gemini, Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. arXiv preprint arXiv:2403.05530, 2024.

[20] Shester Gueuwou, Sophie Siake, Colin Leong, and Mathias Müller. JWSign: A highly multilingual corpus of Bible translations for more diversity in sign language processing. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, Findings of the Association for Computational Linguistics: EMNLP 2023, pages 9907–9927, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.664. URL https://aclanthology.org/2023.findings-emnlp.664.

[21] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. arXiv preprint arXiv:2306.11644, 2023.

[22] Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In Loic Barrault, Ondrej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussa, Christian Federmann, Mark Fishel, Alexander Fraser, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Tom Kocmi, Andre Martins, Makoto Morishita, and Christof Monz, editors, Proceedings of the Sixth Conference on Machine Translation, pages 478–494, Online, November 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.wmt-1.57.

[23] Sneha Kudugunta, Isaac Rayburn Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. MADLAD-400: A multilingual and document-level large audited dataset. In Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2023. URL https://openreview.net/forum?id=Y45ZCxslFx.

[24] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 8424–8445, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.577. URL https://aclanthology.org/2022.acl-long.577.

[25] Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. Gloss-free end-to-end sign language translation. arXiv preprint arXiv:2305.12876, 2023.

[26] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. Mediapipe: A framework for building perception pipelines. arXiv preprint arXiv:1906.08172, 2019.

[27] Qingsong Ma, Johnny Wei, OndÅ™ej Bojar, and Yvette Graham. Results of the wmt19 metrics shared task: Segment-level and strong mt systems pose big challenges. In Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1), pages 62–90, Florence, Italy, August 2019. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W19-5302.

[28] Mathias Müller, Malihe Alikhani, Eleftherios Avramidis, Richard Bowden, Annelies Braffort, Necati Cihan Camgöz, Sarah Ebling, Cristina España-Bonet, Anne Göhring, Roman Grundkiewicz, Mert Inan, Zifan Jiang, Oscar Koller, Amit Moryossef, Annette Rios, Dimitar Shterionov, Sandra Sidler-Miserez, Katja Tissi, and Davy Van Landuyt. Findings of the second WMT shared task on sign language translation (WMT-SLT23). In Philipp Koehn, Barry Haddow, Tom Kocmi, and Christof Monz, editors, Proceedings of the Eighth Conference on Machine Translation, pages 68–94, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.wmt-1.4. URL https://aclanthology.org/2023.wmt-1.4.

[29] Thao Nguyen, Gabriel Ilharco, Mitchell Wortsman, Sewoong Oh, and Ludwig Schmidt. Quality not quantity: On the interaction between dataset design and robustness of clip. Advances in Neural Information Processing Systems, 35:21455–21469, 2022.

[30] Team OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. arXiv preprint arXiv:2303.08774, 2023.

[31] 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 on Association for Computational Linguistics, ACL '02, page 311–318, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://doi.org/10.3115/1073083.1073135.

[32] Maja Popović. chrF: character n-gram F-score for automatic MT evaluation. In Ondřej Bojar, Rajan Chatterjee, Christian Federmann, Barry Haddow, Chris Hokamp, Matthias Huck, Varvara Logacheva, and Pavel Pecina, editors, Proceedings of the Tenth Workshop on Statistical Machine Translation, pages 392–395, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-3049. URL https://aclanthology.org/W15-3049.

[33] Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, et al. Scaling speech technology to 1,000+ languages. arXiv preprint arXiv:2305.13516, 2023.

[34] Amy Pu, Hyung Won Chung, Ankur P Parikh, Sebastian Gehrmann, and Thibault Sellam. Learning compact metrics for mt. In Proceedings of EMNLP, 2021.

 https://doi.org/10.52202/079017-3621

[35] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research, 21(1):5485–5551, 2020.

[36] Adam Roberts, Hyung Won Chung, Gaurav Mishra, Anselm Levskaya, James Bradbury, Daniel Andor, Sharan Narang, Brian Lester, Colin Gaffney, Afroz Mohiuddin, et al. Scaling up models and data with t5x and seqio. Journal of Machine Learning Research, 24(377): 1–8, 2023.

[37] Phillip Rust, Bowen Shi, Skyler Wang, Necati Cihan Camgöz, and Jean Maillard. Towards privacy-aware sign language translation at scale. arXiv preprint arXiv:2402.09611, 2024.

[38] Marcelo Sandoval-Castaneda, Yanhong Li, Bowen Shi, Diane Brentari, Karen Livescu, and Gregory Shakhnarovich. TTIC's submission to WMT-SLT 23. In Philipp Koehn, Barry Haddow, Tom Kocmi, and Christof Monz, editors, Proceedings of the Eighth Conference on Machine Translation, pages 344–350, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.wmt-1.35. URL https://aclanthology.org/2023.wmt-1.35.

[39] Noam Shazeer and Mitchell Stern. Adafactor: Adaptive learning rates with sublinear memory cost. In International Conference on Machine Learning, pages 4596–4604. PMLR, 2018.

[40] Bowen Shi, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. Open-domain sign language translation learned from online video. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 6365–6379, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.427. URL https://aclanthology.org/2022.emnlp-main.427.

[41] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, Advances in Neural Information Processing Systems, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf.

[42] Garrett Tanzer. Fleurs-asl: Including american sign language in massively multilingual multitask evaluation, 2024. URL https://arxiv.org/abs/2408.13585.

[43] Garrett Tanzer. Fingerspelling within sign language translation, 2024. URL https://arxiv.org/abs/2408.07065.

[44] Garrett Tanzer and Biao Zhang. Youtube-sl-25: A large-scale, open-domain multilingual sign language parallel corpus, 2024. URL https://arxiv.org/abs/2407.11144.

[45] Laia Tarrés, Gerard I. Gállego, Amanda Duarte, Jordi Torres, and Xavier Giró i Nieto. Sign language translation from instructional videos, 2023.

[46] 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, 2024.

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017.

[48] Andreas Voskou, Konstantinos P Panousis, Harris Partaourides, Kyriakos Tolias, and Sotirios Chatzis. A new dataset for end-to-end sign language translation: The greek elementary school dataset. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1966–1975, 2023.

[49] Zirui Wang, Zihang Dai, Barnabas Poczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.

[50] 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.

[51] Baixuan Xu, Haochen Shi, Tianshi Zheng, Qing Zong, Weiqi Wang, Zhaowei Wang, and Yangqiu Song. KnowComp submission for WMT23 sign language translation task. In Philipp Koehn, Barry Haddow, Tom Kocmi, and Christof Monz, editors, Proceedings of the Eighth Conference on Machine Translation, pages 351–358, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.wmt-1.36. URL https://aclanthology.org/2023.wmt-1.36.

[52] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. arXiv preprint arXiv:2010.11934, 2020.

[53] Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. ByT5: Towards a token-free future with pre-trained byte-to-byte models. Transactions of the Association for Computational Linguistics, 10:291–306, 2022. doi: 10.1162/tacl_a_00461. URL https://aclanthology.org/2022.tacl-1.17.

[54] Jinhui Ye, Wenxiang Jiao, Xing Wang, Zhaopeng Tu, and Hui Xiong. Cross-modality data augmentation for end-to-end sign language translation. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, Findings of the Association for Computational Linguistics: EMNLP 2023, pages 13558–13571, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.904. URL https://aclanthology.org/2023.findings-emnlp.904.

[55] Aoxiong Yin, Zhou Zhao, Weike Jin, Meng Zhang, Xingshan Zeng, and Xiaofei He. Mlslt: Towards multilingual sign language translation. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 5099–5109, 2022. doi: 10.1109/CVPR52688.2022.00505.

[56] Biao Zhang and Rico Sennrich. Edinburgh's end-to-end multilingual speech translation system for IWSLT 2021. In Marcello Federico, Alex Waibel, Marta R. Costa-jussà, Jan Niehues, Sebastian Stuker, and Elizabeth Salesky, editors, Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021), pages 160–168, Bangkok, Thailand (online), August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.iwslt-1.19. URL https://aclanthology.org/2021.iwslt-1.19.

[57] Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. Improving massively multilingual neural machine translation and zero-shot translation. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1628–1639, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.148. URL https://aclanthology.org/2020.acl-main.148.

[58] 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.

[59] 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, pages 20871–20881, 2023.

[60] 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, pages 1316–1325, 2021.

[61] Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. Spatial-temporal multi-cue network for sign language recognition and translation. IEEE Transactions on Multimedia, 24:768–779, 2021.

 https://doi.org/10.52202/079017-3621

A Appendix

Table 7: Data statistics for YouTube sign language and MADLAD spoken language. We list ISO 639-3 code, language name, and the number of hours/clips/videos for sign language; for spoken language, we list BCP-47 code, language name and the number of parallel examples. "K": thousand, "M": million. Note that like Tanzer and Zhang [44] pre-filtering and Tanzer [43], these language labels are heuristically estimated based on public video metadata, such as caption language and text in the video title, description, etc.

Sign Language (SL)					Spoken Language		
ISO 639-3	Name	# Hours	# Clips	# Videos	BCP-47	Name	# Examples
ase	American SL	2.8K	285.2K	25.9K	es	Spanish	292.8M
bzs	Brazilian SL	590.4	60.2K	5.9K	de	German	283.3M
pso	Polish SL	421.8	41.8K	3.2K	fr	French	243.6M
ins	Indian SL	375.3	39.3K	5.7K	it	Italian	100.1M
bfi	British SL	267.3	27.4K	2.7K	cs	Czech	53.1M
gsg	German SL	235.2	24.2K	2.1K	pl	Polish	42.9M
fsl	French SL	193.7	19.9K	2.9K	ru	Russian	29.0M
jsl	Japanese SL	176.5	17.9K	3.0K	zh	Simplified Chinese	25.9M
ise	Italian SL	161.0	16.8K	2.4K	ar	Arabic	18.2M
asf	Australian SL	123.1	12.5K	1.5K	ja	Japanese	5.3M
rsl	Russian SL	119.4	12.1K	1.2K	hi	Hindi	1.2M
csc	Catalan SL	114.8	11.8K	1.7K	nl	Dutch	93.1M
csn	Colombian SL	107.8	11.1K	478.0	pt	Portuguese	83.7M
aed	Argentine SL	86.1	8.8K	522.0	sv	Swedish	51.9M
mfs	Mexican SL	76.9	7.6K	434.0	hu	Hungarian	40.0M
kvk	Korean SL	69.3	7.1K	612.0	da	Danish	38.2M
hsh	Hungarian SL	55.9	5.8K	1.3K	fi	Finnish	34.1M
sgg	Swiss German SL	48.7	5.1K	634.0	el	Greek	25.2M
prl	Peruvian SL	42.3	4.2K	147.0	sk	Slovak	25.0M
fse	Finnish SL	41.2	4.3K	593.0	no	Norwegian	19.4M
swl	Swedish SL	38.7	4.0K	563.0	bg	Bulgarian	15.5M
asq	Austrian SL	31.8	3.2K	592.0	lt	Lithuanian	15.3M
tsm	Turkish SL	31.8	3.3K	446.0	lv	Latvian	14.3M
dse	Dutch SL	31.4	3.2K	352.0	sl	Slovenian	11.8M
cse	Czech SL	29.2	3.0K	336.0	et	Estonian	11.0M
inl	Indonesian SL	27.5	2.8K	236.0	ko	Korean	5.8M
nsl	Norwegian SL	22.0	2.2K	215.0	hr	Croatian	5.3M
hks	Hong Kong SL	21.3	2.2K	248.0	is	Icelandic	4.1M
tss	Taiwan SL	19.6	2.0K	258.0	sr	Serbian	2.5M
gss	Greek SL	18.2	1.9K	169.0	tr	Turkish	2.5M
dsl	Danish SL	16.0	1.6K	188.0	vi	Vietnamese	1.5M
csg	Chilean SL	15.2	1.5K	184.0	id	Indonesian	1.4M
sfb	French Belgian SL	14.4	1.5K	358.0	he	Hebrew	1.1M
isr	Israeli SL	14.3	1.4K	289.0	th	Thai	1.1M
vietnam	Vietnamese SL	14.2	1.1K	97.0	ms	Malay	907.5K
isg	Irish SL	13.2	1.4K	101.0	uk	Ukrainian	881.0K
slovenia	Slovenian SL	12.7	1.3K	177.0	ca	Catalan	686.2K
nzs	New Zealand SL	12.3	1.3K	224.0	ta	Tamil	396.9K
icl	Icelandic SL	11.1	1.2K	213.0	fil	Filipino	369.8K
sls	Singapore SL	9.9	1.0K	148.0	ne	Nepali	277.9K
tsq	Thai SL	9.0	923.0	150.0	cy	Welsh	93.3K
pks	Pakistani SL	8.6	902.0	145.0			
svk	Slovak SL	8.5	886.0	114.0			
jos	Jordanian SL	8.3	880.0	124.0			
lls	Lithuanian SL	7.6	792.0	166.0			
csr	Costa Rican SL	7.6	789.0	45.0			
psr	Portuguese SL	7.4	764.0	127.0			
rms	Romanian SL	7.3	747.0	60.0			
xml	Malaysian SL	7.2	668.0	85.0			
ecs	Ecuadorian SL	7.2	727.0	28.0			
psp	Filipino SL	6.8	715.0	85.0			
sfs	South African SL	5.2	543.0	41.0			
ugy	Uruguay SL	4.7	483.0	113.0			

esn	Salvadoran SL	3.9	403.0	15.0
xki	Kenyan SL	3.8	384.0	23.0
serbia	Serbian SL	3.3	351.0	36.0
csq	Croatian SL	3.1	326.0	31.0
esl	Egyptian SL	3.0	262.0	32.0
psl	Puerto Rican SL	1.8	181.0	17.0
bengladesh	Bengali SL	1.6	165.0	15.0
gsm	Guatemalan SL	1.4	144.0	26.0
xms	Moroccan SL	1.2	125.0	4.0
lsp	Panamanian SL	1.2	123.0	8.0
fcs	Quebec SL	0.9	91.0	27.0
eso	Estonian SL	0.9	90.0	13.0
emirati	UAE SL	0.8	81.0	14.0
vsl	Venezuelan SL	0.7	69.0	15.0
pys	Paraguayan SL	0.7	69.0	5.0
kazakh	Kazakh SL	0.6	69.0	6.0
hds	Honduran SL	0.6	67.0	8.0
macau	Macau SL	0.6	109.0	109.0
sdl	Saudi Arabian SL	0.5	49.0	7.0
doq	Dominican SL	0.5	45.0	10.0
belarus	Belarusian SL	0.4	29.0	5.0
bqn	Bulgarian SL	0.3	31.0	9.0
sqs	Sri Lankan SL	0.3	29.0	8.0
lsl	Latvian SL	0.2	27.0	3.0
bvl	Bolivian SL	0.2	26.0	5.0
nsi	Nigerian SL	0.1	12.0	2.0
nsp	Nepali SL	0.1	6.0	1.0

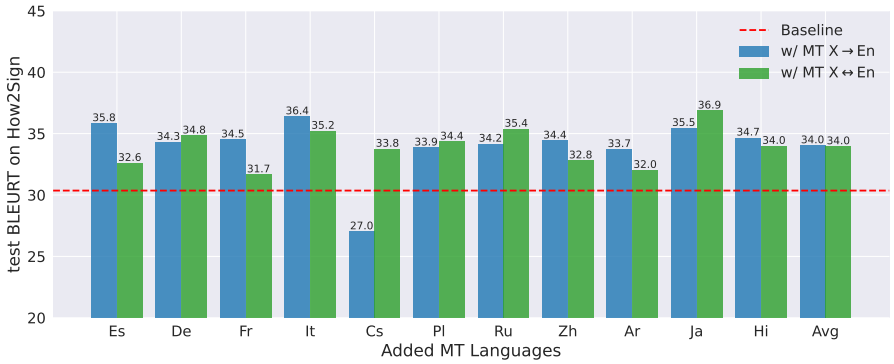


Figure 5: Pretraining performance for *Baseline + MT* when varying MT languages on How2Sign test set. We show BLEURT↑ results and set $p_{mt} = 0.5$. Note only YT-ASL and bilingual MT data are used, i.e. MT languages are added separately instead of jointly. Results are for ByT5 Base. "X→En": MT data for translation into English; "X↔En": MT data for both translation directions; "Avg": average performance over languages. MT languages are arranged in descending order from left to right based on the quantity of translation data available for each language.

A.1 Setup

Sign Video Preprocessing Our landmark preprocessing is identical to [46], and we use the same random 34-second video clipping as [42]. We preprocess sign language video with its default frame rate but discard every second frame for computational efficiency. We convert each video frame to a 255-dimensional normalized vector using MediaPipe Holistic landmarks [26], which also facilitates video anonymization. The input video is eventually transformed into a vector sequence and then mapped to the encoder via a linear projection layer.

Downstream Benchmarks Note the official SLT track in WMT23 for LIS-CH and LSF-CH is for sign language generation rather than translation. We reversed it as a SLT dataset. FLEURS-ASL#0 is the subset of FLEURS-ASL [42] recorded by signer #0, i.e. 353 sentences from FLORES [10]

https://doi.org/10.52202/079017-3621

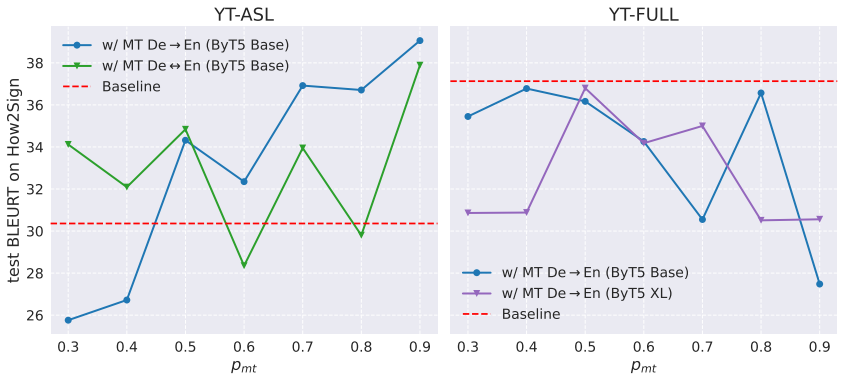


Figure 6: Pretraining performance for *Baseline + MT* when changing the mixing ratio of MT data p_{mt} on How2Sign test set. We show BLEURT↑ results and vary p_{mt} from 0.3 to 0.9. Note only bilingual MT En-De data are explored.

translated into ASL by a Certified Deaf Interpreter. We report only signer #0 because the rest of the benchmark was not complete when these experiments were run.

For these benchmarks, we only use the sign language video and target text *without* glosses. All SLT models in this study are gloss-free.

Model Setting For **Pretraining**, we use a batch size of 256 and a constant learning rate of 0.001. We pretrain models up to 1M steps using 64/64/128 TPU-v3 chips for Base/Large/XL, taking 7-20 days. We select the best checkpoint for downstream application based on the How2Sign dev performance measured by BLEU[5].

For **Finetuning**, we use a batch size of 32 and a constant learning rate of 0.0005. By default, we perform finetuning on each downstream benchmark separately. We only consider the SLT task at finetuning, and directly finetune the model on well aligned (sign video segment, target translation) pairs, which is provided in all downstream benchmarks. We tune models up to 50K steps using 16/32 TPU-v3 chips for Base/XL, taking 2∼5 days. We select the best checkpoint for final evaluation based on the dev-set BLEU.

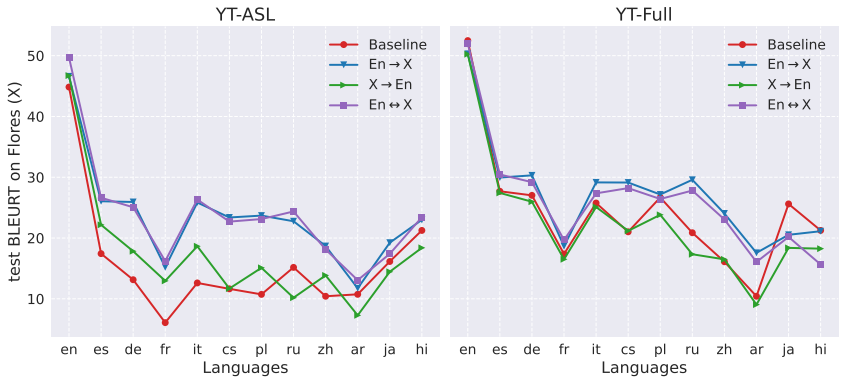
A.2 More Results and Analysis

Zero-shot SLT benefits from MT data partially because it reduces off-target translation. A key factor affecting zero-shot MT is the off-target problem, where the model translates into a wrong target language [57]. We examine this problem for zero-shot SLT according to experiments in Figure 3b and show the results in Table 10.

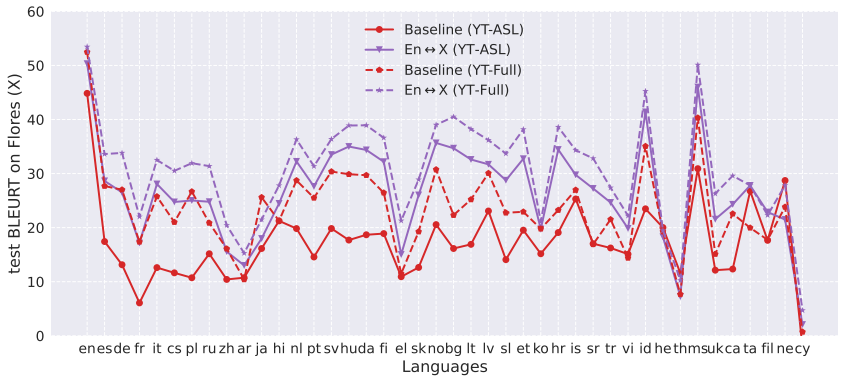
We noticed that zero-shot SLT also suffers from off-target translation, particularly for those languages distant from English. For example, Baseline only has a language accuracy of 1.7, 45.0 and 42.2 for Cs, Zh and Hi, respectively. Adding MT data generally improves the translation language accuracy, such as 1.7/78.5 to 61.2/97.7 for Cs/Pl. But there are also exceptions, like Zh and Hi, where the accuracy reduces from 45.0/42.2 to 15.0/4.8. A deeper inspection shows that jointly training with MT leads to more empty outputs for these languages: the empty rate increases from 2.8/1.4 to 17.3/43.3 for Zh/Hi. We argue that this may be because 1) these languages have significantly less parallel MT data, e.g. Hi only has 1.2M examples, and 2) the parallel corpus from MADLAD can also be quite noisy.

Different evaluation metrics may disagree. There is a hot debate in MT community regarding which metric we should use for translation evaluation [22]. While BLEU has been widely adopted, it often shows poor correlation with human evaluation, particularly when the translation models are strong [27]. Instead, neural metrics are recommended [17]. We follow this trend and adopt

[5]We didn't adopt BLEURT for model selection because it's significantly more expensive and time-consuming than BLEU.



(a) Results for training with MT-Small.



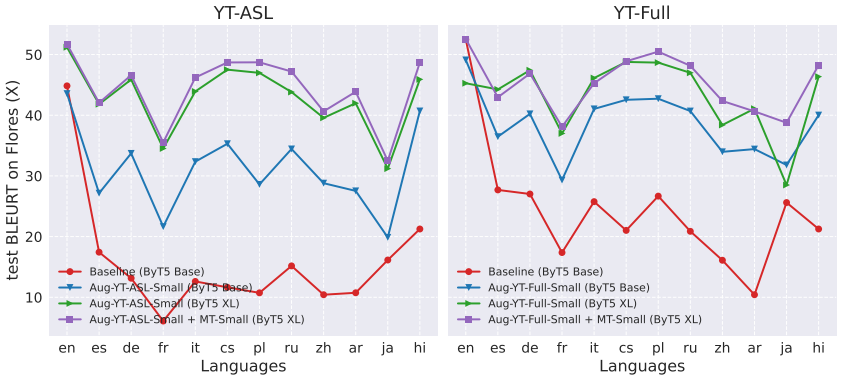
(b) Results for training with MT-Large.

Figure 7: Per-language pretraining performance for *Baseline + MT* with $p_{mt} = 0.9$ when scaling up languages and data. We show BLEURT↑ results on FLEURS-ASL#0. We add multilingual MT data into SLT pretraining and compare MT-Small with MT-Large. Results are for ByT5 Base.

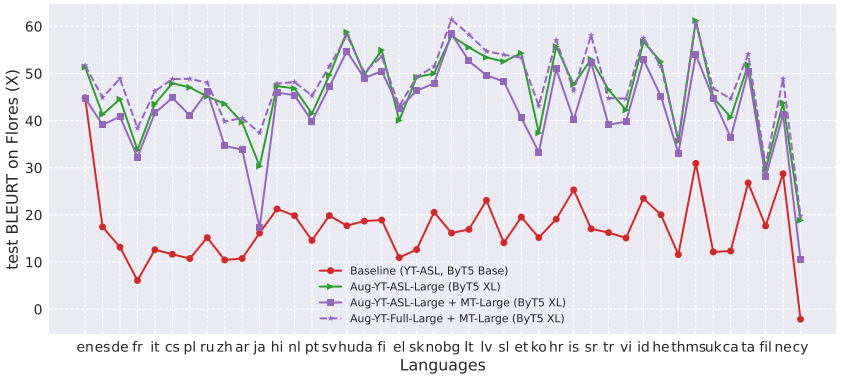
BLEURT as the main metric. To be compatible with past studies and also ease future comparison, we also add BLEU and ChrF. Table 8 shows some disagreements between BLEURT and BLEU/ChrF. For example, model (4) performs better than (comparable to) model (5) on average based on ChrF (BLEU), while BLEURT scores show a clear superiority of model (5) over (4). Evaluation metric selection should be more careful due to these disagreements. In this study, we rely more on BLEURT for the analysis as it correlates better with human evaluation [17].

Qualitative examples of translation quality. See Table 11 for qualitative examples of finetuned ASL to English translation on How2Sign from our best model, compared to the prior state of the art set by Tanzer [43]. Our model resolves some small issues in (2) and (3) analyzed in Appendix C of Tanzer [43]; the remaining discrepancies in these particular examples seem to be issues with the dataset quality.

https://doi.org/10.52202/079017-3621



(a) Results for training with MT-Small and Aug-YT-ASL/Full-Small.



(b) Results for training with MT-Large and Aug-YT-ASL/Full-Large.

Figure 8: Per-language pretraining performance for SLT with augmented SLT data. We show BLEURT↑ results for *Baseline + Augmented SLT + MT* with $p_{mt} = 0.9$ on FLEURS test set. MT data are multilingual in both directions. Data augmentation substantially improves SLT performance across languages.

https://doi.org/10.52202/079017-3621

ID	Model	H2S	E23	WMT23				Avg
				LIS-CH	LSF-CH	SRF	SS	
0	Prevous SOTA	18.10	5.69	5.20	7.00	0.30	7.50	7.30
1	ByT5 Base	3.71	7.68	0.40	0.79	1.01	3.10	2.78
2	1 + Baseline + YT-ASL	17.94	16.58	6.59	8.65	2.14	8.69	10.10
3	2 + MT-Small ($p_{mt} = 0.9$)	17.30	17.90	8.84	11.86	2.04	11.01	11.49
4	3 + Aug-YT-ASL-Small	18.55	22.09	9.81	12.91	2.90	14.34	13.43
5	4 + Aug-YT-ASL&MT-Large + ByT5 XL	19.31	24.08	9.70	11.71	2.76	13.26	13.47
6	2 + YT-Full	19.79	21.81	12.99	15.32	2.07	13.71	14.28
7	6 + Aug-YT-ASL&MT-Small	18.98	23.60	13.63	15.75	2.85	14.44	14.88
8	7 + Aug-YT-ASL&MT-Large + ByT5 XL	**21.06**	**25.65**	14.93	**18.77**	2.80	**18.17**	**16.90**
9	8 + Multilingual SLT Tuning	19.25	23.05	**16.79**	17.22	**2.91**	15.92	15.86

(a) BLEU↑ scores.

ID	Model	H2S	E23	WMT23				Avg
				LIS-CH	LSF-CH	SRF	SS	
0	Prevous SOTA	-	-	-	-	17.50	-	-
1	ByT5 Base	19.55	25.19	14.81	15.56	11.08	21.75	17.99
2	1 + Baseline + YT-ASL	38.78	41.34	27.57	29.99	17.06	38.31	31.18
3	2 + MT-Small ($p_{mt} = 0.9$)	39.01	45.40	30.56	33.67	16.13	40.38	34.19
4	3 + Aug-YT-ASL-Small	39.64	47.92	31.40	34.95	17.49	43.56	35.83
5	4 + Aug-YT-ASL&MT-Large + ByT5 XL	40.15	48.70	29.44	33.20	17.96	40.21	34.94
6	2 + YT-Full	41.13	47.71	38.23	38.13	16.78	42.33	37.38
7	6 + Aug-YT-ASL&MT-Small	40.35	49.57	38.07	39.88	19.34	42.83	38.34
8	7 + Aug-YT-ASL&MT-Large + ByT5 XL	**41.97**	**50.09**	38.90	**40.43**	**19.58**	**45.94**	**39.49**
9	8 + Multilingual SLT Tuning	40.39	48.49	**40.28**	38.63	18.72	45.22	38.62

(b) ChrF↑ scores.

ID	Model	H2S	E23	WMT23				Avg
				LIS-CH	LSF-CH	SRF	SS	
0	Prevous SOTA	50.80	-	25.20	18.80	24.60	37.70	-
1	ByT5 Base	34.00	22.14	22.77	7.74	15.41	26.88	21.49
2	1 + Baseline + YT-ASL	51.74	37.79	24.24	15.43	21.82	35.59	31.10
3	2 + MT-Small ($p_{mt} = 0.9$)	52.62	45.98	33.10	24.58	23.33	45.45	37.51
4	3 + Aug-YT-ASL-Small	53.36	49.34	38.61	28.70	25.87	49.61	40.91
5	4 + Aug-YT-ASL&MT-Large + ByT5 XL	54.28	54.16	38.93	27.29	28.42	51.73	42.47
6	2 + YT-Full	53.51	49.48	42.11	31.16	21.15	44.28	40.28
7	6 + Aug-YT-ASL&MT-Small	53.70	53.13	45.09	37.69	30.31	52.45	45.40
8	7 + Aug-YT-ASL&MT-Large + ByT5 XL	**55.69**	**56.94**	51.94	**41.14**	**33.94**	57.96	**49.60**
9	8 + Multilingual SLT Tuning	53.47	55.57	**54.54**	39.26	29.33	**58.08**	48.38

(c) BLEURT↑ scores.

Table 8: Finetuning performance on downstream SLT benchmarks. "H2S/E23": How2Sign/Elementary23. "SRF/SS": WMT23 DSGS SRF/SS test split. "Avg": averaged performance over all benchmarks. MT data are added in both translation directions. Previous SOTA: How2Sign [43], Elementary23 [48] and WMT23 SRF [28], WMT23 LIS-CH, LSF-CH, SS [44]. Scaling SLT reaches new SOTA across benchmarks. *All models are finetuned on each SLT benchmark separately except (9).*

https://doi.org/10.52202/079017-3621

ID	Model	H2S	E23	WMT23				Avg
				LIS-CH	LSF-CH	SRF	SS	
1	ByT5 Base							
2	1 + Baseline + YT-ASL	3.77	0.06	0.15	0.35	0.15	0.15	0.77
3	2 + MT-Small ($p_{mt} = 0.9$)	4.75	0.02	0.07	0.25	0.06	0.12	0.88
4	3 + Aug-YT-ASL-Small	3.31	0.01	0.12	0.43	0.12	0.31	0.72
5	4 + Aug-YT-ASL&MT-Large + ByT5 XL	2.81	0.21	0.22	0.31	0.06	0.24	0.64
6	2 + YT-Full	5.78	0.33	3.43	5.69	1.08	3.88	3.37
7	6 + Aug-YT-ASL&MT-Small	4.10	0.05	1.67	2.65	0.50	2.21	1.86
8	7 + Aug-YT-ASL&MT-Large + ByT5 XL	4.05	2.45	4.50	3.73	0.64	3.45	3.14

(a) BLEU↑ scores.

ID	Model	H2S	E23	WMT23				Avg
				LIS-CH	LSF-CH	SRF	SS	
1	ByT5 Base							
2	1 + Baseline + YT-ASL	20.65	6.62	12.67	15.1	13.89	13.95	13.81
3	2 + MT-Small ($p_{mt} = 0.9$)	19.55	0.05	10.9	11.33	10.75	12.89	10.91
4	3 + Aug-YT-ASL-Small	15.21	0.05	9.67	9.87	5.89	14.28	9.16
5	4 + Aug-YT-ASL&MT-Large + ByT5 XL	11.40	9.25	8.75	6.88	3.35	7.55	7.86
6	2 + YT-Full	23.44	14.81	25.34	26.78	16.42	28.36	22.53
7	6 + Aug-YT-ASL&MT-Small	18.18	10.22	19.81	22.64	10.25	25.95	17.84
8	7 + Aug-YT-ASL&MT-Large + ByT5 XL	13.47	22.34	25.53	26.16	14.82	26.96	21.55

(b) ChrF↑ scores.

ID	Model	H2S	E23	WMT23				Avg
				LIS-CH	LSF-CH	SRF	SS	
1	ByT5 Base							
2	1 + Baseline + YT-ASL	30.36	9.13	9.32	6.33	9.69	10.45	12.55
3	2 + MT-Small ($p_{mt} = 0.9$)	34.24	1.07	16.38	10.44	13.14	14.81	15.01
4	3 + Aug-YT-ASL-Small	25.2	1.63	23.41	10.82	10.65	22.06	15.61
5	4 + Aug-YT-ASL&MT-Large + ByT5 XL	23.87	10.35	21.58	6.75	7.04	15.96	14.26
6	2 + YT-Full	37.13	15.08	28.92	18.33	17.87	29.66	24.50
7	6 + Aug-YT-ASL&MT-Small	25.25	12.68	33.54	24.48	10.66	33.91	23.42
8	7 + Aug-YT-ASL&MT-Large + ByT5 XL	22.41	34.14	43.07	34.26	21.52	39.47	32.48

(c) BLEURT↑ scores.

Table 9: Pretraining performance on downstream SLT benchmarks.

Language	es	de	fr	it	cs	pl	ru	zh	ar	ja	hi
Baseline	90.9	85.6	92.4	82.4	1.7	78.5	95.2	45.0	72.2	76.5	42.2
w/ MT X↔En	95.2	93.2	94.1	91.2	61.2	97.7	85.3	15.0	75.4	73.7	4.8

(a) Language Accuracy: the accuracy of translations in the correct target language.

Language	es	de	fr	it	cs	pl	ru	zh	ar	ja	hi
Baseline	8.2	4.0	6.5	5.1	7.6	2.3	0.0	2.8	5.1	0.3	1.4
w/ MT X↔En	2.0	0.0	0.6	0.6	0.6	0.0	7.4	17.3	18.7	0.3	43.3

(b) Empty Rate: the proportion of outputting empty translations.

Table 10: Analysis for zero-shot SLT in Figure 3b. Higher language accuracy indicates less off-target translation, thus better quality; lower empty rate is better.

(1)	**Reference**	And that's a great vital point technique for women's self defense.
	Tanzer [43]	It's a really great point for women's self defense.
	Ours	It's a really great point for women's self defense.
(2)	**Reference**	In this clip I'm going to show you how to tape your cables down.
	Tanzer [43]	In this segment I'm going to show you how to draw a name of the cable.
	Ours	In this clip I'm going to show you how to tape the cable.
(3)	**Reference**	In this segment we're going to talk about how to load your still for distillation of lavender essential oil.
	Tanzer [43]	In this clip we're going to talk about how to install a still for ditching nice for a lavender oil.
	Ours	In this clip, we're going to talk about how to feed the still for digestion and for lavender oil.
(4)	**Reference**	You are dancing, and now you are going to need the veil and you are going to just grab the veil as far as possible.
	Tanzer [43]	Now we're going to have her dance, and now we're going to have her braided her hair up as far as possible.
	Ours	Your dancing, now we need the fringe to come up as far as possible.
(5)	**Reference**	But if you have to setup a new campfire, there's two ways to do it in a very low impact; one is with a mound fire, which we should in the campfire segment earlier and the other way to setup a low impact campfire is to have a fire pan, which is just a steel pan like the top of a trash can.
	Tanzer [43]	But if you have to set up a new campfire, there's two ways to do it in a low impact, one is a bond fire, which we should do in a campfire stack early and the other one is to set up a campfire in a fire pan, that's just a steel pan like the top of the pan.
	Ours	But if you have to set up a new campfire, there's two ways to do it in a low impact, one is a bond fire, which we should do in campfire sanding early, and the other one is to set up a campfire in a fire pan, which is just a steel pan like the tops of the pans.
(6)	**Reference**	So, this is a very important part of the process.
	Tanzer [43]	This is a part of the process.
	Ours	And that's okay, part of the process.

Table 11: **Qualitative examples of finetuned sentence-level ASL to English translation results on How2Sign**, instances originally selected by Tarrés et al. [45]. We compare the reference, the prior SOTA Tanzer [43], and our best model.

https://doi.org/10.52202/079017-3621

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We summarize our contributions in the introduction, which are supported by empirical results in the Section 4.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discussed the limitations in Section 6 and the Ethics Statement.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our study builds on top of empirical experiments without theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We elaborate the pretraining and finetuning algorithm in Section 2 with optimization and evaluation details in Section 3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

https://doi.org/10.52202/079017-3621

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification:

In terms of data: MADLAD-400 is publicly available, but our noisy YouTube dataset is not. However, we provide information about the crawling method, and the covered sign languages and data statistics. The results could also be reproduced directionally with YouTube-SL-25, which is a smaller but cleaner dataset.

In terms of code: The used (m/By)T5 model checkpoints are publicly available, but the framework we used to finetune them with multimodal inputs has not been open sourced, so we are unable to release our code. However, the methodology is simple and we provide the details to replicate it.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide training and test details in Section A.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to high computational cost, we run SLT pretraining and finetuning once for each setting. To make sure the evaluation is reliable, we compare different methods with several metrics, including BLEU, ChrF and BLEURT.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide details for the computational resources in Section 3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our work follows the ethics guidelines of NeurIPS 2024.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

https://doi.org/10.52202/079017-3621

Justification: We discussed potential societal impacts in Section 6

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We didn't release data or models from this study.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The used data and models are properly cited in this paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We didn't release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our study doesn't include crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our study doesn't include crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

 https://doi.org/10.52202/079017-3621

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.