information

Article

Leveraging Frozen Pretrained Written Language Models for Neural Sign Language Translation

Mathieu De Coster * and Joni Dambre

IDLab-AIRO, Ghent University—IMEC, Technologiepark-Zwijnaarde 126, 9052 Ghent, Belgium; joni.dambre@ugent.be
* Correspondence: mathieu.decoster@ugent.be

Abstract: We consider neural sign language translation: machine translation from signed to written languages using encoder–decoder neural networks. Translating sign language videos to written language text is especially complex because of the difference in modality between source and target language and, consequently, the required video processing. At the same time, sign languages are low-resource languages, their datasets dwarfed by those available for written languages. Recent advances in written language processing and success stories of transfer learning raise the question of how pretrained written language models can be leveraged to improve sign language translation. We apply the Frozen Pretrained Transformer (FPT) technique to initialize the encoder, decoder, or both, of a sign language translation model with parts of a pretrained written language model. We observe that the attention patterns transfer in zero-shot to the different modality and, in some experiments, we obtain higher scores (from 18.85 to 21.39 BLEU-4). Especially when gloss annotations are unavailable, FPTs can increase performance on unseen data. However, current models appear to be limited primarily by data quality and only then by data quantity, limiting potential gains with FPTs. Therefore, in further research, we will focus on improving the representations used as inputs to translation models.

Keywords: sign language translation; machine translation; transfer learning

Citation: De Coster, M.; Dambre, J. Leveraging Frozen Pretrained Written Language Models for Neural Sign Language Translation. *Information* **2022**, *13*, 220. https://doi.org/10.3390/info13050220

Academic Editor: Willy Susilo

Received: 15 March 2022
Accepted: 22 April 2022
Published: 23 April 2022

1. Introduction

Machine translation (MT) from signed to written languages is a two-stage task. First, sign language videos have to be processed to some salient representation. This stage is commonly referred to as sign language recognition (SLR). Then, this representation is used as input to an MT model that performs the actual translation to written language text. MT is increasingly becoming more powerful for written languages, and advances in the domain are also being applied for sign language translation (SLT).

Datasets used for MT from signed to written languages are significantly smaller than typical datasets used for MT between different written languages. The most popular dataset for translation from sign language video to written language text is RWTH-PHOENIX-Weather 2014T [1]. This dataset consists of 8257 parallel utterances between German sign language (in video format) and German (in text format). In contrast, the ParaCrawl corpora contain up to several million sentence pairs [2]. The most common MT models today are neural MT models, based on deep neural networks. These are primarily designed for MT between pairs of written languages, more often than not high-resource languages. These models are therefore optimized to work with large quantities of data and do not scale well to low-resource languages as they are. This needs to be taken into account when these models are adapted for SLT.

The lack of labeled data for MT can be mitigated in several ways. One possibility is the generation of synthetic training data through data augmentation [3] or back-translation [4,5]. Another possibility—explored in this article as an extension of previous research—is to leverage language models that are pretrained on larger corpora [6]. Here, instead of training neural translation models from scratch, the encoder, decoder, or both, are replaced

by a pretrained language model. This transfer learning, when applied in the direction of high-resource to low-resource language pairs, can improve the performance for those low-resource language pairs [7]. These language models do not necessarily need to be pretrained specifically for translation. They can also be pretrained on monolingual corpora for the purpose of transfer learning. Models such as the well-known "BERT" [8] that are trained in this way can also be adapted as encoders or decoders in an MT model to improve translation performance [9]. Such models can even transfer in few or zero-shot to different languages [10,11], but also across modalities [12], and this gives them great potential in low-resource scenarios. Even if the downstream task is not related to natural language processing, incorporating pretrained language models can provide a powerful initialization for model parameters and improve the performance on the downstream task. Again, this is possible without additional training of the attention mechanism, i.e., in a few- or zero-shot setting [13]. Using a (partly) frozen pretrained written language model to initialize a model for a different downstream task is called the "Frozen Pretrained Transformers" (FPT) technique.

In written language translation, sequences of discrete tokens are used. These tokens typically correspond to words or parts of words. Signed languages, unlike written languages, have no standardized written form. Therefore, many papers on SLT use frame based or clip based representations of sign language videos as inputs to the translation model [1,5,6,14,15]. These representations are typically longer than sequences of words. Furthermore, such representations often exhibit a large degree of correlation between neighbors in the sequence (e.g., between two subsequent frames in a video).

The fact that pretrained language models can be used for tasks that are unrelated to natural language processing [13] suggests that the attention patterns learned by large language models generalize to other tasks and modalities. This raises the question of whether the attention patterns also transfer to the longer, more intra-correlated, sequences used in SLT. It turns out that they do. By integrating (a small number of layers of) BERT into a sign language machine translation model, one can match and even exceed the performance of models trained from scratch in terms of the BLEU-4 translation score [6]. The FPT approach to SLT is illustrated in Figure 1.

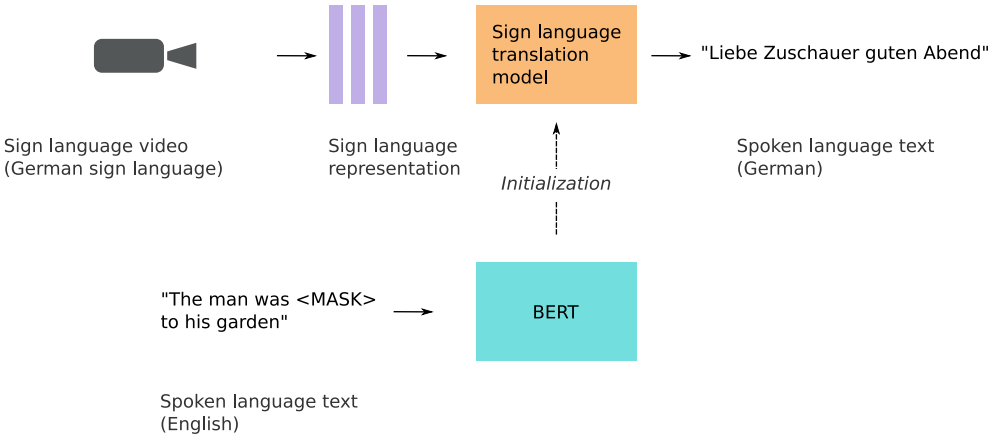

Figure 1. In our proposed approach to SLT, pretrained written language models are integrated into the SLT pipeline as initialization for (parts of) the translation model, and then partly frozen and partly fine-tuned on the translation task [1].

We further explore FPTs for SLT in this extension article (of [6]). We focus on the cases where the encoder of a transformer based translation model is replaced by BERT or where encoder and decoder are replaced by BERT. We show that the gains in performance are not due to the model architecture, but indeed due to pretraining on unrelated written language corpora. We furthermore extend the analysis beyond reporting the BLEU-4 score: we also report the ROUGE-L and CHRF scores of our models on unseen data. We additionally perform several qualitative analyses on FPTs to gain insights into the technique.

We showcase some example translations of transformers trained from scratch and FPTs. Learning curves show that the models we are investigating are primarily limited by data quality rather than data quantity. Finally, we compare models with gloss level supervision (Sign2(Gloss+Text)) and models without gloss level supervision (Sign2Text). We show that an FPT trained without glosses can match the performance of a baseline model trained with glosses.

2. Background

2.1. Sign Language Machine Translation

SLT (from a signed to a written language) is a video-to-text translation task. Typically, sign language utterances (as parts of conversations or monologues) are recorded as videos, and these videos need to be translated into written language utterances with the same meaning. As sign languages have no standardized written form, one needs to extract relevant information from these videos to feed to the translation model. Many papers translate from sign language glosses to written language text (e.g., Refs. [1,15–19]). These glosses form a transcription of the sign language video with written language words or clauses. This method is similar to written language MT: they both translate from text to text. However, glosses do not convey the full meaning of what is being signed [20,21] and they are often influenced by written language [20,22]. Hence, end-to-end SLT (from video rather than glosses), is required to fully extract the meaning of the sign language utterances. Due to practical hardware limitations, the videos need to be converted into feature vectors before training the translation model: therefore, a powerful offline feature extractor is required. The design of this feature extractor is typically based on SLR techniques. The feature vectors are used as inputs to MT models, which borrow their architecture from written language MT models.

Early SLT models used statistical MT techniques (e.g., Refs. [16–19]). More recently, neural MT techniques have become the standard for SLT (e.g., Refs. [1,5,6,14,15,23–25]). Advances in MT and in computer vision have made this end-to-end MT from sign language video to written language text possible. The first neural Sign2Text model was introduced together with the RWTH-PHOENIX-Weather 2014T dataset that we consider in this article. The recurrent encoder–decoder model with Luong attention [26] and achieves a BLEU-4 score of 9.58 on the test set [1]. Camgoz et al. [1] argue that glosses are more salient representations than what can be extracted using convolutional neural networks and therefore they also propose a Sign2Gloss2Text model. In such a model, sign language glosses are first predicted by a continuous SLR (CSLR) model and then these glosses are translated into written language text. This simplifies the translation task and can benefit from research into SLR, which, while still unsolved, is more mature than SLT. By using this technique, they achieve higher scores: 18.13 BLEU-4 on the same test set.

However, translating from sign language glosses to written language text is not desirable. Glosses do not convey the full meaning of signing [20,22] and may present an information bottleneck [14]. Nevertheless, they can still provide useful information to bootstrap the learning process of a sign language translation system. When used as a side channel rather than a bottleneck, they can provide an additional supervised signal to improve translation performance. A Sign2(Gloss+Text) translation model performs CSLR and SLT jointly, but the translation input is the sign language representation as in a Sign2Text model and not glosses as in a Sign2Gloss2Text model. The glosses are instead added as a side channel to be predicted based on the sign language inputs. This has several benefits: glosses no longer are an information bottleneck for the translation model, and gloss level annotations are not required for all training data, nor do glosses need to be predicted during inference. The original Sign2(Gloss+Text) model uses transformers [14] instead of recurrent neural networks and achieves a BLEU-4 test score of 21.32 on the RWTH-PHOENIX-Weather 2014T dataset.

Pretrained language models for many written languages are frequently made publicly available to the scientific community. The Hugging Face Model Hub serves over thirty-

five thousand models at the time of writing: https://huggingface.co/models (accessed on 21 April 2022). These pretrained models can be used for downstream tasks such as text summarization and question answering, but also to initialize translation models. In our previous work, we used frozen pretrained written language models for SLT on the RWTH-PHOENIX-Weather 2014T dataset in a Sign2(Gloss+Text) set-up [6]. We observe a potential performance improvement of 1 to 2 BLEU-4 when integrating frozen pretrained transformers.

2.2. Frozen Pretrained Transformers

The parameters of transformers that are trained on large language datasets can be transferred to downstream tasks that are not related to language [13]. In fact, it is possible to freeze the attention computation modules in these transformers, as the attention patterns learned in the pretraining setting transfer in zero-shot to downstream tasks. This requires only minor modifications to the transformer architecture: a linear input layer must be added to transform the new type of inputs to align with the attention modules and a linear output layer is required depending on the downstream task. Then, the pretrained transformer can be fine-tuned on the downstream task.

Pretraining on language tasks yields better performance on the downstream task than random initialization or pretraining on related tasks with less data: this suggests that the quantity of pretraining data offsets the difference in the data modality.

The original paper on FPTs uses GPT-2 [27], a generative language model. A linear input layer is added, and the positional embeddings as well as the layer normalization parameters are fine-tuned, while other parameters are frozen. FPTs converge faster than models trained from scratch and only a fraction of the layers of the pretrained model suffices to outperform random initialization. In some cases, additionally fine-tuning the feedforward layer parameters can improve performance. BERT [8] is also evaluated, and similar results are obtained when compared to GPT-2.

2.3. Related Work

Pretrained language models provide a warm start for training neural networks on multiple downstream tasks such as question answering and MT, for written languages.

Imamura and Sumita [28] integrate a BERT checkpoint in the encoder of their machine translation model. They use a two-stage training approach, where first only the decoder is trained (with a frozen BERT encoder), and then both encoder and decoder are fine-tuned. They show that integrating BERT increases the effectiveness of the machine translation model in low-resource settings. Rothe et al. [9] not only initializes the encoder of the MT model, but also the decoder. They evaluate the effects of initializing MT models with BERT, GPT-2 and RoBERTa [29] checkpoints. They employ these checkpoints in several locations and combinations, e.g., RoBERTa in the encoder and GPT-2 in the decoder, or BERT in the encoder and the decoder. They obtain the best results with BERT-based models, outperforming a baseline trained from scratch.

Pretrained language models can even be leveraged across language boundaries. Artetxe et al. [10] show that the attention patterns learned by BERT on one written language transfer to another written language with minimal fine-tuning. Gogoulou et al. [11] further illustrate how semantic information is transferred between different languages.

Written language model checkpoints have also been leveraged in the domain of sign language translation. Miyazaki et al. [30] present a machine translation model from written text to sign language glosses. They augment the written language encoder with a pretrained language model.

The above papers use a written language model to initialize an encoder and/or decoder that processes written language text. The novelty of our work specifically lies in the transfer of a written language model (BERT) to a sign language encoder: a BERT model is first pretrained on English text and it is then used to encode German Sign Language utterances. These utterances are not represented as text. They are modeled as sequences of

feature vectors which are extracted frame per frame in the sign language video. In other words, we show that the attention patterns of BERT trained on written text transfer in zero shot to this different language, and more importantly, to this different modality.

3. Materials and Methods

3.1. Sign Language Representation

Due to GPU memory constraints, we are limited to offline feature extraction to obtain sign language representations from sign language videos. We use the same representation as in our previous work on FPTs for SLT [6], which is also the same as the representation used by the model that we consider to be our baseline [14]. This representation is thus preextracted for the entire dataset.

The feature extractor is a 2D Convolutional Neural Network (CNN) that extracts frame-wise embedding vectors: i.e., every frame in a video is transformed into a corresponding feature vector. This 2D CNN is an Inception network [31] pretrained as part of a larger model for CSLR.

This larger model combines a CNN with a Long Short-Term Memory (LSTM) network and a Hidden Markov Model (HMM). This model was originally used for weakly supervised learning of SLR [32]. The CNN-LSTM combination performs labeling, while the HMM performs alignment between labels and video frames. A detailed explanation of the feature extractor is out of scope for this article; more information can be found in the original paper [32].

We extract the CNN from the pretrained CNN-LSTM-HMM model and apply it frame-wise on the RWTH-PHOENIX-Weather 2014T dataset to obtain one feature vector for every frame. Sequences of these feature vectors are the inputs to the encoder of our encoder–decoder translation model.

3.2. Sign Language Transformers

We transform a baseline transformer based translation model into an FPT based model. The baseline, taken from the pioneering work on SLT with transformers [14], has three layers in both encoder and decoder. We refer to this model as *Baseline*.

As a first modification to the baseline model, we replace the encoder with the first two layers of BERT. This number of layers was found to be optimal in previous research [6]. We use the weights of the BERT model trained by its original authors [8]. Hence, BERT is pretrained to perform masked language modeling and next sentence prediction on English text. We refer to this model architecture as *BERT2RND*.

We also perform experiments where the encoder and the decoder are both replaced with the first two layers of BERT: we refer to this model as *BERT2BERT*. We use two fine-tuning schemes, as in our previous work. In the first scheme, we fine-tune only the layer normalization parameters. In the second scheme, we fine-tune layer normalization parameters and feedforward layer parameters (the original work on FPTs showed that fine-tuning feedforward layer parameters can yield better performance [13]). The difference is indicated in superscript: $BERT2RND^{ln}$ and $BERT2BERT^{ln}$ for the first scheme and $BERT2RND^{ff}$ and $BERT2BERT^{ff}$ for the second. In order to determine whether any increase in translation scores is due to the architecture of BERT or due to the pretraining on monolingual English texts, we also consider $BERT2RND^{scratch}$ and $BERT2BERT^{scratch}$. These models have the same BERT based architecture, but they are trained entirely from scratch on the sign language data (i.e., there is no transfer learning). In summary, we consider the following models:

- Baseline
- $\text{BERT2RND}^{\text{scratch}}$
- $\text{BERT2BERT}^{\text{scratch}}$
- $\text{BERT2RND}^{\text{ff}}$
- $\text{BERT2BERT}^{\text{ff}}$
- $\text{BERT2RND}^{\text{ln}}$
- $\text{BERT2BERT}^{\text{ln}}$

3.3. Sign Language Translation

We consider two approaches to SLT. Firstly, we look at end-to-end translation in the form of a Sign2Text approach. The sign language representation is the input of the encoder, and the output of the decoder is the sequence of written language words. Secondly, we provide additional supervision to the encoder through the form of glosses. The model jointly performs CSLR and SLT, by predicting glosses from the sign language context. That is, we add an auxiliary classifier on top of the encoder and train it using Connectionist Temporal Classification (CTC) [33]. This approach, which is called Sign2(Gloss+Text), has been empirically shown to improve translation performance [14]. Figure 2 illustrates the Sign2Text and Sign2(Gloss+Text) network architectures.

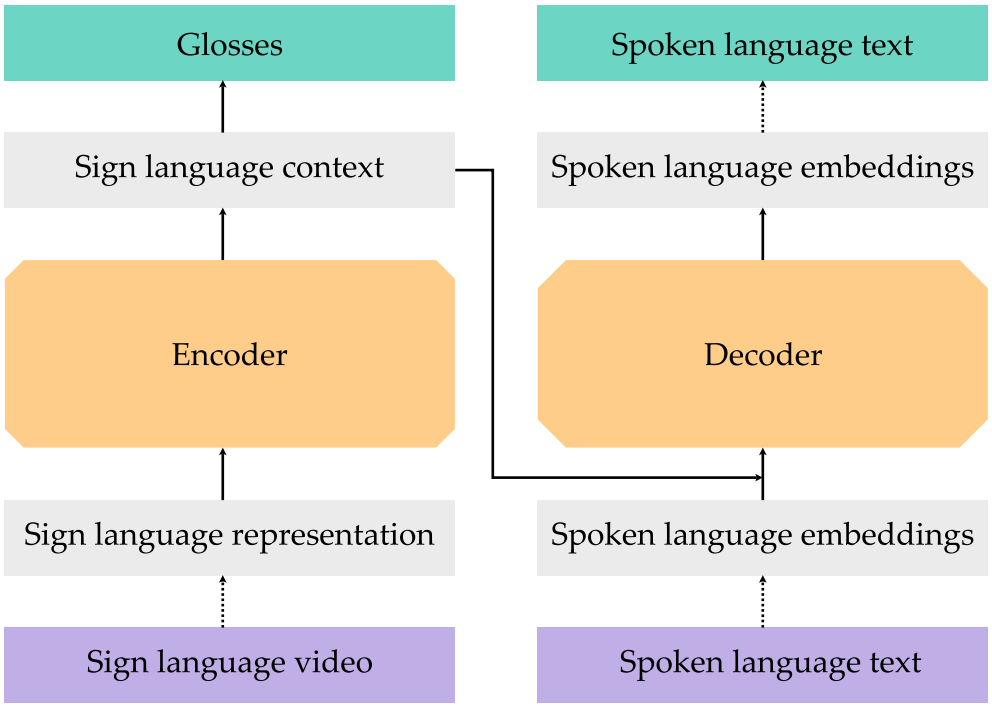

Figure 2. A neural SLT model translates from a sign language video to a written language text using an encoder–decoder model. Solid arrows indicate transformations that are trained end-to-end. Dashed arrows indicate an offline conversion: for example, the sign language videos are converted to sign language representations before training the SLT model. The gloss output (top left) is only present in a Sign2(Gloss+Text) setup.

3.4. Implementation Details

To construct the BERT2RND models from the Transformer baseline, we replace the three encoder layers with two layers from the BERT-base [8] model. Two layers were shown to be optimal for BERT2BERT$^{\text{ff}}$, BERT2BERT$^{\text{ln}}$ and BERT2RND$^{\text{ff}}$ in our previous experiments [6]. While the BLEU-4 score was slightly higher with only one layer for BERT2RND$^{\text{ln}}$, the difference was negligible: we therefore choose two layers for all models to have comparable complexity. We freeze all parameters in multi-head attention layers except the layer normalization parameters in BERT2RND$^{\text{ln}}$. For BERT2RND$^{\text{ff}}$, we additionally fine-tune the feedforward layer parameters. For BERT2RND$^{\text{scratch}}$, we train all parameters from scratch on the sign language dataset: we transfer the BERT architecture, but not the pretrained weights.

The BERT2BERT models are constructed similarly, by replacing both encoder and decoder using a BERT model. BERT—an encoder-only model designed for the processing of monolingual data—only features self-attention, but a decoder in a translation model also requires cross-attention. Therefore, we add a cross-attention layer that is trained from scratch (it is not frozen in any of the variants, as that would imply that random attention patterns would be used).

As per previous research [6], all models are optimized using Adam [34], with a batch size of 32. We use weight decay (0.001) to reduce overfitting. We decrease the learning rate by a factor of 0.7 whenever the development set BLEU-4 score has not increased for 800 iterations. We stop training when the learning rate is smaller than 1×10^{-7}.

Previous research suggests that the choice of a random seed may have a significant impact on model performance [6]. We account for this variance by using five arbitrarily chosen random seeds (1, 93, 2021, 251016 and 7366756) and running every experiment for each seed. We then report average scores and the standard deviation.

Our experiments are built on the open source PyTorch [35] code base from "Frozen Pretrained Transformers for Neural Sign Language Translation" [6], which is based on SignJoey [14], in its turn derived from JoeyNMT [36]. We provide the source code that can be used to reproduce all of our experiments on GitHub (https://github.com/m-decoster/fpt4slt) (accessed on 21 April 2022).

3.5. Evaluation

We measure the following translation scores: BLEU-4 [37], which corresponds to translation precision, ROUGE-L [38], which can be interpreted as the translation recall, and CHRF [39], the 6-gram F-score, with recall weighted twice as high as precision. We also perform a qualitative analysis by providing example translations of the different models.

4. Results

4.1. Experimental Results

Table 1 provides an overview of the scores on the development set for the different models. Table 2 shows the same overview for the test set. We can observe several patterns in these results, which we discuss in the following sections.

4.2. Performance Comparison

In order to determine whether or not FPTs can be used to improve the performance of SLT models, we compare BLEU-4, ROUGE-L and CHRF scores for the Baseline model with the BERT2RND^{ff}, BERT2RND^{ln}, BERT2BERT^{ff} and BERT2BERT^{ln} models. Figure 3 shows this comparison.

For Sign2Text, we observe a clear difference between the Baseline model and the FPTs: on average, FPTs outperform the Baseline model in terms of BLEU-4, ROUGE-L and CHRF. This can not be said for Sign2(Gloss+Text), where the difference is less pronounced and the BERT2BERT^{ln} model performs worse than the Baseline model. This is in line with findings from previous research, suggesting that the decoder of the translation model benefits from having more degrees of freedom to model the written language text [6].

Note that these models have a different architecture from the Baseline model. Therefore, we investigate whether the performance gains for Sign2Text are due to the architecture or due to the pretraining on a monolingual (English) text corpus.

Table 1. Overview of the scores on the *development* set for the different models, averaged over five trials with different random initialization.

Task	Model	BLEU-4	ROUGE-L	CHRF
Sign2Text	Baseline	19.09 ± 0.59	44.18 ± 0.25	41.35 ± 0.75
	BERT2RND$^{\text{scratch}}$	19.67 ± 0.75	44.02 ± 1.35	42.00 ± 1.12
	BERT2RND$^{\text{ff}}$	21.58 ± 0.41	47.36 ± 0.56	44.44 ± 0.73
	BERT2RND$^{\text{ln}}$	20.95 ± 0.51	46.46 ± 0.62	43.78 ± 0.45
	BERT2BERT$^{\text{scratch}}$	17.35 ± 1.26	41.25 ± 1.74	39.11 ± 1.51
	BERT2BERT$^{\text{ff}}$	20.77 ± 0.55	46.83 ± 0.72	43.93 ± 0.42
	BERT2BERT$^{\text{ln}}$	19.92 ± 0.45	46.03 ± 0.72	42.77 ± 0.47
Sign2(Gloss+Text)	Baseline	20.68 ± 0.46	46.39 ± 0.64	43.79 ± 0.70
	BERT2RND$^{\text{scratch}}$	21.95 ± 0.49	47.63 ± 0.83	44.44 ± 0.50
	BERT2RND$^{\text{ff}}$	21.97 ± 0.65	47.54 ± 0.69	44.69 ± 0.78
	BERT2RND$^{\text{ln}}$	20.64 ± 0.24	46.37 ± 0.39	43.12 ± 0.43
	BERT2BERT$^{\text{scratch}}$	21.34 ± 0.73	47.17 ± 1.00	43.88 ± 0.89
	BERT2BERT$^{\text{ff}}$	21.44 ± 0.69	47.41 ± 0.51	44.33 ± 0.99
	BERT2BERT$^{\text{ln}}$	19.77 ± 0.39	46.19 ± 0.59	42.77 ± 0.48

Table 2. Overview of the scores on the *test* set for the different models, averaged over five trials with different random initializations.

Task	Model	BLEU-4	ROUGE-L	CHRF
Sign2Text	Baseline	18.85 ± 0.69	44.01 ± 0.70	41.69 ± 0.91
	BERT2RND$^{\text{scratch}}$	19.45 ± 0.61	43.79 ± 1.20	42.36 ± 1.02
	BERT2RND$^{\text{ff}}$	21.39 ± 0.63	46.67 ± 0.63	44.66 ± 0.77
	BERT2RND$^{\text{ln}}$	20.74 ± 0.58	46.24 ± 0.82	43.84 ± 0.63
	BERT2BERT$^{\text{scratch}}$	17.24 ± 1.75	41.15 ± 2.53	39.74 ± 2.19
	BERT2BERT$^{\text{ff}}$	21.07 ± 0.80	46.49 ± 0.79	44.41 ± 0.69
	BERT2BERT$^{\text{ln}}$	19.98 ± 0.35	46.02 ± 0.72	43.02 ± 0.47
Sign2(Gloss+Text)	Baseline	20.18 ± 0.64	45.59 ± 1.13	43.78 ± 0.81
	BERT2RND$^{\text{scratch}}$	21.82 ± 0.22	47.25 ± 0.32	45.05 ± 0.24
	BERT2RND$^{\text{ff}}$	21.52 ± 0.59	47.00 ± 0.67	44.82 ± 0.70
	BERT2RND$^{\text{ln}}$	20.62 ± 0.73	46.27 ± 0.66	43.66 ± 0.76
	BERT2BERT$^{\text{scratch}}$	21.36 ± 0.84	47.04 ± 0.88	44.44 ± 0.94
	BERT2BERT$^{\text{ff}}$	21.23 ± 0.88	47.22 ± 0.64	44.76 ± 0.94
	BERT2BERT$^{\text{ln}}$	19.44 ± 0.63	45.71 ± 0.70	42.87 ± 0.30

4.3. Model Architecture

Any potential performance gain from using the BERT architecture, compared to using the baseline architecture, can be observed by comparing the BERT2RND$^{\text{scratch}}$ and BERT2BERT$^{\text{scratch}}$ models with the Baseline model. In Figure 4, there appears to be no difference between *scratch* and the Baseline model for Sign2Text, suggesting that the choice of architecture (between BERT or Baseline) has little impact on the performance of the models on unseen data.

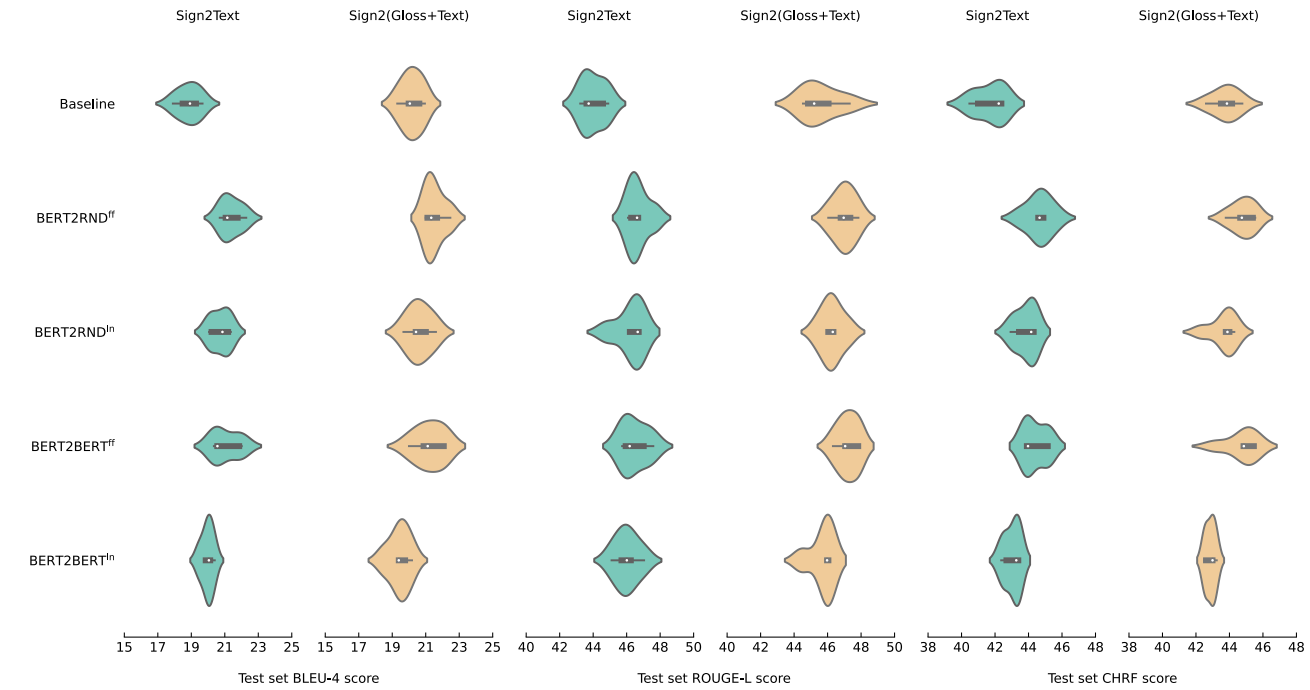

Figure 3. The FPT models outperform the Baseline model for the Sign2Text task. For the Sign2(Gloss+Text) task, the difference is less pronounced.

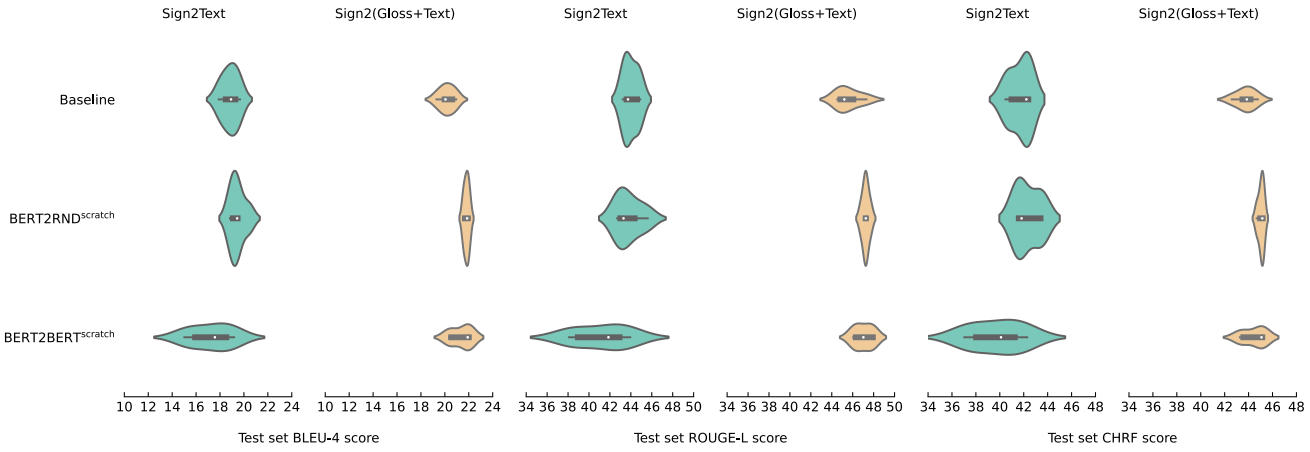

Figure 4. For Sign2Text, using the BERT2RND or BERT2BERT architecture does not yield performance gains. For Sign2(Gloss+Text), we observe slight performance gains compared to the Baseline model.

4.4. Glosses

In the previous sections, we noted that the models trained from scratch (Baseline, BERT2RND$^{\text{scratch}}$ and BERT2BERT$^{\text{scratch}}$) benefit from the additional gloss information, while the FPTs do not benefit as much or at all.

From these results, we can draw two conclusions from different points of view. Firstly, by using FPTs, we can close the gap between Sign2Text and Sign2(Gloss+Text) performance. Secondly, using FPTs has little benefit when we train the translation model with additional gloss level supervision. In other words, in situations where no gloss level annotations are available, FPTs can be beneficial to the model's translation performance.

We can reconcile these points of view via the following explanation of this phenomenon. We first remark that every one of these models suffers from overfitting on the

training data (as will be illustrated in the next section). Then, we remark that adding the auxiliary supervision in the form of glosses with the Sign2(Gloss+Text) training task regularizes the model. We observe that using FPTs also introduces a form of regularization: the attention patterns learned from written language texts transfer to SLT so that we no longer need to train the attention modules and therefore we can freeze the majority of the pretrained transformer's parameters to regularize the model. We see diminishing returns on the benefits of regularization when these two forms (gloss information and FPTs) are combined.

4.5. Learning Curves

We generate learning curves by combining the median models (see Appendix A) of all model types. We use 25%, 50%, 75% and 100% of the training set to train a model and plot the BLEU-4 performance on the training set and development set in function of the training set size.

Figure 5 shows the learning curves for Sign2Text on the left and Sign2(Gloss+Text) on the right. Figure 6 zooms in on the development set curve and combines the curves for Sign2Text and Sign2(Gloss+Text) in a single graph. Here, we clearly observe (1) that all models are heavily overfitting to the training set, (2) that Sign2(Gloss+Text) models tend to perform better on unseen data on average, and (3) that the slope of the development set curve is flattening as the size of the training set increases.

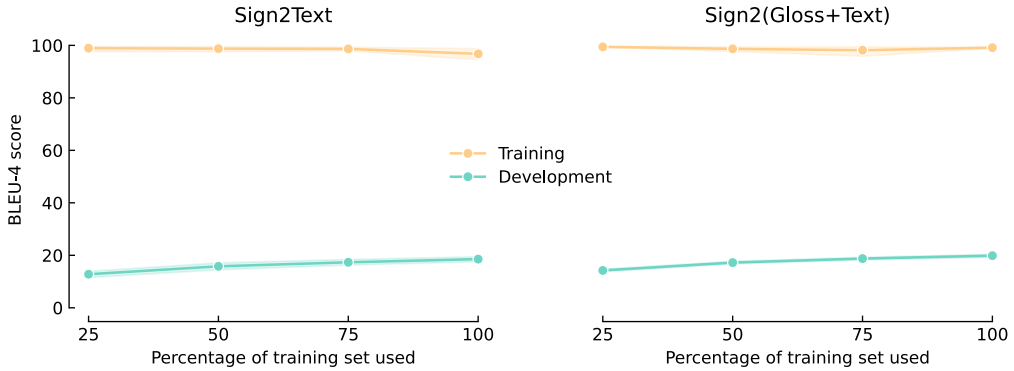

Figure 5. The learning curves for all Sign2Text models (**left**) and all Sign2(Gloss+Text) models (**right**) show clear overfitting. They also show that adding additional data would not yield large improvements to the BLEU-4 scores of the models.

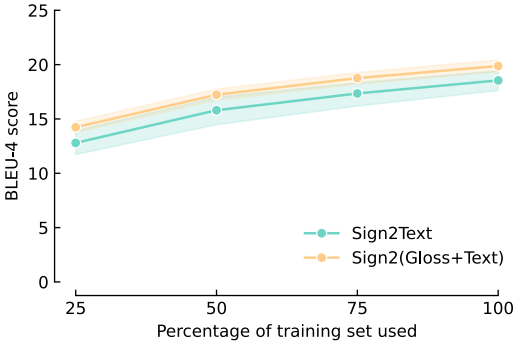

Figure 6. Zooming in on the development set curves of Figure 5, we observe that Sign2(Gloss+Text) models outperform Sign2Text models on average, and that they have lower variance. The slopes, however, are similar.

This implies that the current models, while they are heavily overfitting, will not benefit majorly from additional training data. Instead, the primary factor limiting performance is not the *quantity* of the data, but rather the *quality*. Data cleaning and the extraction of more

salient representations for the sign language videos are required to further close the gap between training and development set scores.

4.6. Example Translations

The best performing FPT, in terms of the average development set BLEU-4 score, according to Table 1, is BERT2RND^{ff} for Sign2Text and Sign2(Gloss+Text). We take the median BERT2RND^{ff} models (see Appendix A) for these two tasks. We then extract three reference translations and the hypotheses of the models. We also do this for the Baseline model. The three reference translations are taken at random from the test set (with indices 114, 25 and 281 generated randomly with seed 42). We compare these translations qualitatively, to determine whether there is any clearly observable difference in the quality of the translations between the best performing FPT and the Baseline. The translations are shown in Tables 3–5. We provide all generated translations for all models, as well as reference translations, in Supplementary Material (S1).

In Table 3, we see an example of our models failing to correctly translate the sentence. The word "freundlich" (friendly) occurs in the translations, but the meaning of the forecast is missed. Some models predict sunny weather (Sign2Text Baseline) while others predict rain (Sign2(Gloss+Text) Baseline) or wind (Sign2(Gloss+Text) BERT2RND^{ff}). Geographical areas are missed entirely. Note that "biskaya" only occurs five times in the training set and "shetlandinseln" does not occur in the training set at all. Table 4 shows examples of translations that are not grammatically correct, but convey the main information in the weather forecast: rain and thunder. Finally, Table 5 shows perfect translations for one type of sentence: the announcement of the day which will be discussed next in the weather forecast. These sentences have a rigid structure that is repeated throughout the entire dataset, making them easy to translate. All of the trained models can correctly translate this example.

Table 3. Example hypotheses generated by the median BERT2RND^{ff} model and the Baseline model. English translations are obtained using Google Translate (1 out of 3).

Model	**Hypothesis**
Example: 03July_2011_Sunday_tagesschau-1665 Reference: in den übrigen landesteilen wird es meist freundlich dank einer hochdruckzone die von der biskaya bis zu den shetlandinseln reicht English translation: in the remaining parts of the country it is mostly friendly thanks to a high pressure zone that stretches from the bay of biscay to the shetland islands	
Sign2Text	
Baseline	sonst klart es verbreitet auf und es wird später auf trotzdem wird es recht freundlich und es fällt dort bis in tiefe lagen *otherwise it clears up widely and it gets up later, nevertheless it becomes quite friendly and it drops down to low levels there*
BERT2RND^{ff}	im übrigen land wird es verbreitet freundlich die macht sich morgen dann in den kommenden tagen von der ems bis nach brandenburg *in the rest of the country it is spreading friendly which will then make itself felt tomorrow in the coming days from the ems to brandenburg*
Sign2(Gloss+Text)	
Baseline	sonst wird es deutlich freundlicher und erreicht das hoch nur einen recht freundliches wetter in den kommenden tagen mit den regenwolken *otherwise it will be much friendlier and the high will only be quite friendly weather in the coming days with the rain clouds*
BERT2RND^{ff}	sonst wird es meist freundlich und von westen weht ein schwacher bis mäßiger wind aus unterschiedlichen richtungen *otherwise it will mostly be friendly and a weak to moderate wind will blow from the west from different directions*

Table 4. Example hypotheses generated by the median BERT2RND^{ff} model and the Baseline model. English translations are obtained using Google Translate (2 out of 3).

Model	Hypothesis
Example: 12July_2010_Monday_tagesschau-374 Reference: morgen gibt es im osten und südosten bei einer mischung aus sonne und wolken zum teil kräftige schauer oder gewitter English translation: tomorrow there will be some heavy showers or thunderstorms in the east and southeast with a mixture of sun and clouds	
Sign2Text	
Baseline	morgen im osten und südosten noch sommerliche werte am nachmittag einzelne schauer und gewitter *tomorrow in the east and southeast some showers and thunderstorms in the afternoon*
BERT2RND^{ff}	morgen im osten und südosten zunächst noch freundlich sonst viele wolken und zum teil kräftige gewittrige regenfälle *tomorrow in the east and south-east initially still friendly otherwise lots of clouds and partly heavy thundery rain*
Sign2(Gloss+Text)	
Baseline	morgen im osten und südosten noch zweistellige regenfälle sonst teils wolkig oder zum teil heftige gewitter *tomorrow in the east and south-east there will still be double-digit rainfall, otherwise partly cloudy or partly heavy thunderstorms*
BERT2RND^{ff}	morgen im osten und südosten noch teilweise gewittrige schauer *partly thundery showers in the east and southeast tomorrow*

Table 5. Example hypotheses generated by the median BERT2RND^{ff} model and the Baseline model. English translations are obtained using Google Translate (3 out of 3).

Model	Hypothesis
Example: 02December_2009_Wednesday_tagesschau-4039 Reference: und nun die wettervorhersage für morgen donnerstag den dritten dezember English translation: and now the weather forecast for tomorrow thursday the third of december	
Sign2Text	
Baseline	und nun die wettervorhersage für morgen donnerstag den dritten dezember *and now the weather forecast for tomorrow thursday the third of december*
BERT2RND^{ff}	und nun die wettervorhersage für morgen donnerstag den dritten dezember *and now the weather forecast for tomorrow thursday the third of december*
Sign2(Gloss+Text)	
Baseline	und nun die wettervorhersage für morgen donnerstag den dritten dezember *and now the weather forecast for tomorrow thursday the third of december*
BERT2RND^{ff}	und nun die wettervorhersage für morgen donnerstag den dritten dezember *and now the weather forecast for tomorrow thursday the third of december*

Overall, we observe that sentences with rigid reoccurring patterns are translated correctly by our models. We see that general concepts such as rain or friendly weather are present in the models' translations, but that output translations can be grammatically incorrect or can fail to convey the full message. Geographical information appears to be more challenging: region names are in the minority in the dataset or are even completely unseen to the models.

5. Discussion

5.1. Discussion of Results

Current SLT datasets, for example the RWTH-PHOENIX-Weather 2014T dataset considered in this article, are small compared to datasets for written language translation. Additionally, SLT is complex especially because sign languages have no standardized written form: we are required to extract sign language representations from videos. This extraction is part of the unsolved domain of SLR. This lack of data, as well as the complex task of feature extraction, means that SLT researchers are working with small and noisy datasets. As a result, current translation models suffer from severe overfitting to the training data.

To reduce the problem complexity, we leverage pretrained written language models. We show that the self-attention patterns of a pretrained (on English text) BERT-base model transfer in a zero-shot situation to SLT, both in an encoder and a decoder setting, even if we only use two layers of BERT-base. We furthermore show that the gains are not due to the different architecture, but rather stem from the pretraining task. As already shown in previous research, training with additional gloss level information as an auxiliary task can improve the performance of SLT models. However, in the absence of gloss level annotations, applying FPTs can augment the performance by a comparable amount. The application of FPTs can therefore be especially beneficial in cases where gloss level annotations are not available or hard to obtain.

We further observe, by charting learning curves, that additional data will likely not improve model performance for the RWTH-PHOENIX-Weather 2014T dataset. Instead, we suggest investigating the effects of data cleaning and researching more powerful feature extraction techniques.

We also perform a limited qualitative analysis, by presenting three example translations for the baseline and the best performing FPT variant. We see that both models can grasp the general meaning of the message but do not yield accurate translations when considering complete sentences. Here, we do not observe a notable difference in translation quality between the different models. The models especially have difficulty with low-frequency signs and out-of-vocabulary signs: this indicates an important challenge for future work.

Our research is limited in the fact that we only evaluated FPTs on a single dataset and a single sign language representation. Future research may investigate whether FPTs have the same regularization benefit for different data and different sign language representations.

5.2. Representation Power of Neural Sign Language Translation Models

Neural SLT models are typically written language MT models with minor changes to adapt them to the sign language modality [1,14,15,25]. However, we ask the question: what is the representation power of such models? Understanding how SLT models work is challenging, as they are "black boxes": we can observe the input and output of a model, but its inner workings are difficult to interpret.

We can decompose the complete SLT pipeline into two parts. The first part is related to the information extraction from video: the design of a sign language tokenizer [1]. The tokenizer T transforms a video v into a sequence of N tokens $s = (s_1, s_2, \ldots, s_N)$: $T(v) = s$. s is an approximation of the signed content in the video v. The second part of the SLT pipeline is the translation model R itself, which translates the sign language tokens into written language tokens. It models the conditional probability $p(w|s)$ of the generated written language utterance $w = (w_1, w_2, \ldots, w_M)$ given the sign language utterance s. In other words, the SLT model output is $y = R(s) = R(T(v))$.

The sign language representation s, that is, the output of the tokenizer T, is the input to the translation model. To obtain high quality translations, this representation must be meaningful and model the semantics of the sign language utterance. Orbay and Akarun [23] show that the choice of tokenizer has a significant impact on model performance. The quality of the sign language representation entails a bound on the expressive power of the downstream machine translation model.

In the scientific literature, we find two main approaches towards the design of T [25]. First, there are gloss based SLT models (Gloss2Text and Sign2Gloss2Text). Here, the sign language utterance is modeled as a sequence of sign language glosses. Such a representation models the semantics of the sign language utterances. Second, frame based or clip based representations are morphological representations based in the visual domain (Sign2Text and Sign2(Gloss+Text)). When such representations are used, the task of modelling semantics falls on the sign language encoder (which is part of the translation model). Belinkov et al. [40] show, in their discussion of the representation power of written language neural MT models, that character based models focus primarily on morphology,

whereas sub-word based models focus on semantics. Sign2Text and Sign2(Gloss+Text) models can be likened to character models, whereas Gloss2Text and Sign2Gloss2Text models are similar to sub-word models.

The errors made by deep neural networks can be decomposed into the approximation error, estimation error and optimization error [41]. As both T and R are deep neural networks, they both make approximation, estimation and optimization errors. The performance of the complete SLT pipeline is bounded by the errors introduced by T and by R, and any errors made by T are propagated to R. The simpler the sign language representation, the higher the approximation error of T, and thus of R. In our case, we use a single sign language representation for all experiments, hence T is fixed and we cannot reduce the error term associated with it. Therefore, we must reduce the error term associated with the translation model R.

When a frame based representation is used (as is the case in our experiments), the translation model is tasked with modeling the sign language utterance semantics from the morphological representation. This is a significant challenge, and it increases the potential estimation error of the translation model. Previous research has facilitated this task by adding gloss level supervision through the use of Sign2(Gloss+Text) models [14]: as the encoder is tasked to perform CSLR, i.e., to predict glosses, it learns to model semantics. This is possible (shown by Camgoz et al. [14]), so the frame based representation that we use here is rich enough to extract semantic information from. FPTs provide a different approach to the extraction of this semantic information from the sign language representation, and they do not require gloss annotations. BERT was trained to model semantics with English sub-word units. As Gogoulou et al. [11] show, semantic information can be transferred across language boundaries. FPT based SLT models can leverage this pre-learned semantic information to better model the sign language utterance s.

The errors of T are fixed (because T itself is fixed). T therefore determines a lower bound on the error of the SLT model. Here, we reduce the total error by lessening the error of R: we introduce semantic information into the encoder of the translation model. We can do this without adding additional gloss level information, through the integration of FPTs into the sign language model.

An exhaustive theoretical analysis (e.g., as performed by Qi et al. [41] for vector-to-vector regression) is out of the scope of this article. However, in future research, such an analysis may provide useful information on the amount of research effort required in the design of T and R.

6. Conclusions

We transfer BERT-base, pretrained on an English text corpus, to an SLT task from German sign language video to German text. The self-attention patterns of BERT-base transfer in zero-shot to this new task, modality and language. This application of Frozen Pretrained Transformers (FPTs) improves the performance (in terms of BLEU-4, ROUGE-L and CHRF) of SLT models. When gloss level annotations are available, this improvement is on average 1 BLEU-4 (best model: +1.34) compared to a baseline trained from scratch. When gloss level annotations are not available and the task at hand is end-to-end video-to-text SLT, FPTs widen the gap with the baseline: here, they provide an average increase of 1.95 BLEU-4 (best model: +2.54). Qualitative inspection of the generated translations suggests that the increase in scores does not necessarily correlate with the translation quality. However, our research into the behavior of FPTs provides us with some additional useful insights into the task of SLT. Neither the baseline, nor FPTs, would benefit from additional data as much as they would benefit from improving the quality of the training data. Measures to take include data cleaning and researching improved feature extraction techniques from the domain of SLR. Therefore, in future work, we will focus on reevaluating existing and designing new feature extraction techniques.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/info13050220/s1, Table S1: Hypotheses generated by all considered models for the development set.

Author Contributions: Conceptualization, M.D.C. and J.D.; methodology, M.D.C.; software, M.D.C.; validation, M.D.C.; formal analysis, M.D.C.; investigation, M.D.C.; resources, M.D.C.; data curation, M.D.C.; writing—original draft preparation, M.D.C.; writing—review and editing, M.D.C. and J.D.; visualization, M.D.C.; supervision, J.D.; project administration, M.D.C. and J.D.; funding acquisition, M.D.C. and J.D. All authors have read and agreed to the published version of the manuscript.

Funding: Mathieu De Coster's research is funded by the Research Foundation Flanders (FWO Vlaanderen): file number 77410. Work in this article is part of the SignON project. This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant No. 101017255.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare a possible conflict of interest with one of the guest editors of the special issue "Frontiers in Machine Translation", namely Dimitar Shterionov. The authors collaborate closely with Shterionov in the context of the SignON research project and are co-authors on several published and work-in-progress papers. The authors and Shterionov did not on collaborate on this particular article.

Abbreviations

The following abbreviations are used in this manuscript:

MT	Machine Translation
SLR	Sign Language Recognition
CSLR	Continuous Sign Language Recognition
SLT	Sign Language Translation
FPT	Frozen Pretrained Transformer
BERT	Bidirectional Encoder Representations from Transformers
GPT-2	Generative Pretrained Transformer 2
RND	Random
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory
HMM	Hidden Markov Model

Appendix A

As discussed, the experiments for the various models are repeated with different random initializations, by choosing arbitrary random seeds. We observe in the experiment results that there is nonnegligible variance in the BLEU-4 scores. Therefore, for further analyses, we do not choose the best model from every five runs, but, instead, the median model. We do this by selecting the model with the median BLEU-4 score out of the five runs. The seeds that result in these models, per model type, are listed in Table A1.

Table A1. Random seeds resulting in the model with median BLEU-4 score, for every model-task combination.

Task	Model	Seed
Sign2Text	Baseline	1
	BERT2RND^{scratch}	93
	BERT2RND^{ff}	2021
	BERT2RND^{ln}	1

Table A1. *Cont.*

Task	Model	Seed
Sign2Text	BERT2BERT^{scratch}	7366756
	BERT2BERT^{ff}	251016
	BERT2BERT^{ln}	2021
Sign2(Gloss+Text)	Baseline	93
	BERT2RND^{scratch}	93
	BERT2RND^{ff}	7366756
	BERT2RND^{ln}	93
	BERT2BERT^{scratch}	251016
	BERT2BERT^{ff}	7366756
	BERT2BERT^{ln}	93

References

1. Camgoz, N.C.; Hadfield, S.; Koller, O.; Ney, H.; Bowden, R. Neural sign language translation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7784–7793.
2. Esplà-Gomis, M.; Forcada, M.; Ramírez-Sánchez, G.; Hoang, H.T. ParaCrawl: Web-scale parallel corpora for the languages of the EU. In Proceedings of the MTSummit, Dublin, Ireland, 19–23 August 2019.
3. Moryossef, A.; Yin, K.; Neubig, G.; Goldberg, Y. Data Augmentation for Sign Language Gloss Translation. In Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL), Online, 16–20 August 2021; pp. 1–11.
4. Zhang, X.; Duh, K. Approaching Sign Language Gloss Translation as a Low-Resource Machine Translation Task. In Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL), Online, 16–20 August 2021; pp. 60–70.
5. Zhou, H.; Zhou, W.; Qi, W.; Pu, J.; Li, H. Improving Sign Language Translation with Monolingual Data by Sign Back-Translation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 1316–1325.
6. De Coster, M.; D'Oosterlinck, K.; Pizurica, M.; Rabaey, P.; Verlinden, S.; Van Herreweghe, M.; Dambre, J. Frozen Pretrained Transformers for Neural Sign Language Translation. In Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL), Virtual, 20 August 2021; pp. 88–97.
7. Zoph, B.; Yuret, D.; May, J.; Knight, K. Transfer Learning for Low-Resource Neural Machine Translation. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–4 November 2016; pp. 1568–1575. [CrossRef]
8. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers); Association for Computational Linguistics, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. [CrossRef]
9. Rothe, S.; Narayan, S.; Severyn, A. Leveraging pre-trained checkpoints for sequence generation tasks. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 264–280. [CrossRef]
10. Artetxe, M.; Ruder, S.; Yogatama, D. On the Cross-lingual Transferability of Monolingual Representations. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 4623–4637.
11. Gogoulou, E.; Ekgren, A.; Isbister, T.; Sahlgren, M. Cross-lingual Transfer of Monolingual Models. *arXiv* **2021**, arXiv:2109.07348.
12. Tsimpoukelli, M.; Menick, J.; Cabi, S.; Eslami, S.; Vinyals, O.; Hill, F. Multimodal few-shot learning with frozen language models. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 200–212.
13. Lu, K.; Grover, A.; Abbeel, P.; Mordatch, I. Pretrained transformers as universal computation engines. *arXiv* **2021**, arXiv:2103.05247.
14. Camgoz, N.C.; Koller, O.; Hadfield, S.; Bowden, R. 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, Online, 14–19 June 2020; pp. 10023–10033.
15. Yin, K.; Read, J. Better Sign Language Translation with STMC-Transformer. In Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; pp. 5975–5989.
16. Bungeroth, J.; Ney, H. Statistical sign language translation. In Proceedings of the Workshop on Representation and Processing of Sign Languages, LREC, Citeseer, Lisbon, Portugal, 24–30 May 2004; Volume 4, pp. 105–108.
17. Morrissey, S.; Way, A.; Stein, D.; Bungeroth, J.; Ney, H. Combining data-driven MT systems for improved sign language translation. In Proceedings of the Machine Translation Summit XI, Copenhagen, Denmark, 10–14 September 2007.
18. Stein, D.; Schmidt, C.; Ney, H. Sign language machine translation overkill. In Proceedings of the International Workshop on Spoken Language Translation (IWSLT), Paris, France, 2–3 December 2010.

19. Forster, J.; Schmidt, C.; Koller, O.; Bellgardt, M.; Ney, H. Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In Proceedings of the LREC, Reykjavik, Iceland, 26–31 May 2014; pp. 1911–1916.
20. Frishberg, N.; Hoiting, N.; Slobin, D.I. Transcription. In *Sign Language*; Pfau, R., Steinbach, M., Woll, B., Eds.; De Gruyter Mouton: Berlin, Germany, 2012; pp. 1045–1075. [CrossRef]
21. Vermeerbergen, M.; Leeson, L.; Crasborn, O.A. *Simultaneity in Signed Languages: Form and Function*; John Benjamins Publishing: Amsterdam, The Netherlands, 2007; Volume 281.
22. Vermeerbergen, M. Past and current trends in sign language research. *Lang. Commun.* **2006**, *26*, 168–192. [CrossRef]
23. Orbay, A.; Akarun, L. Neural sign language translation by learning tokenization. In Proceedings of the 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), Buenos Aires, Argentina, 16–20 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 222–228.
24. Zhou, H.; Zhou, W.; Zhou, Y.; Li, H. Spatial-temporal multi-cue network for sign language recognition and translation. *IEEE Trans. Multimed.* **2021**, *24*, 768–779. [CrossRef]
25. De Coster, M.; Shterionov, D.; Van Herreweghe, M.; Dambre, J. Machine Translation from Signed to Spoken Languages: State of the Art and Challenges. *arXiv* **2022**, arXiv:2202.03086.
26. Luong, M.T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1412–1421.
27. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
28. Imamura, K.; Sumita, E. Recycling a pre-trained BERT encoder for neural machine translation. In Proceedings of the 3rd Workshop on Neural Generation and Translation, Hong Kong, China, 4 November 2019; pp. 23–31.
29. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
30. Miyazaki, T.; Morita, Y.; Sano, M. Machine translation from spoken language to Sign language using pre-trained language model as encoder. In Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages: Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives, Marseille, France, 11–16 May 2020; pp. 139–144.
31. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
32. Koller, O.; Camgoz, N.C.; Ney, H.; Bowden, R. Weakly supervised learning with multi-stream CNN-LSTM-HMMs to discover sequential parallelism in sign language videos. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 2306–2320. [CrossRef] [PubMed]
33. Graves, A.; Fernández, S.; Gomez, F.; Schmidhuber, J. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, USA, 25–29 June 2006; pp. 369–376.
34. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015.
35. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; pp. 8024–8035.
36. Kreutzer, J.; Bastings, J.; Riezler, S. Joey NMT: A Minimalist NMT Toolkit for Novices. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations, Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Hong Kong, China, 2019; pp. 109–114. [CrossRef]
37. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–17 July 2002; pp. 311–318.
38. Lin, C.Y.; Och, F.J. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04), Barcelona, Spain, 21–26 July 2004; pp. 605–612.
39. Popović, M. chrF: Character n-gram F-score for automatic MT evaluation. In Proceedings of the Tenth Workshop on Statistical Machine Translation, Lisbon, Portugal, 17–18 September 2015; pp. 392–395.
40. Belinkov, Y.; Durrani, N.; Dalvi, F.; Sajjad, H.; Glass, J. On the linguistic representational power of neural machine translation models. *Comput. Linguist.* **2020**, *46*, 1–52. [CrossRef]
41. Qi, J.; Du, J.; Siniscalchi, S.M.; Ma, X.; Lee, C.H. Analyzing upper bounds on mean absolute errors for deep neural network-based vector-to-vector regression. *IEEE Trans. Signal Process.* **2020**, *68*, 3411–3422. [CrossRef]