ECAI 2025
I. Lynce et al. (Eds.)

doi:10.3233/FAIA251315

Leveraging Large Language Models for Accurate Sign Language Translation in Low-Resource Scenarios

Luana Bulla[a,b], **Gabriele Tuccio**[a,b,*], **Misael Mongiovì**[a,b] **and Aldo Gangemi**[b,c]

[a]University of Catania, Italy
[b]National Research Council - ISTC, Italy
[c]University of Bologna, Italy

Abstract. Translating natural languages into sign languages is a highly complex and underexplored task. Despite growing interest in accessibility and inclusivity, the development of robust translation systems remains hindered by the limited availability of parallel corpora which align natural language with sign language data. Existing methods often struggle to generalize in these data-scarce environments, as the few datasets available are typically domain-specific, lack standardization, or fail to capture the full linguistic richness of sign languages. To address this limitation, we propose Advanced Use of LLMs for Sign Language Translation (AulSign), a novel method that leverages Large Language Models via dynamic prompting and in-context learning with sample selection and subsequent sign association. Despite their impressive abilities in processing text, LLMs lack intrinsic knowledge of sign languages; therefore, they are unable to natively perform this kind of translation. To overcome this limitation, we associate the signs with compact descriptions in natural language and instruct the model to use them. We evaluate our method on both English and Italian languages using SignBank+, a recognized benchmark in the field, as well as the Italian LaCAM CNR-ISTC dataset. We demonstrate superior performance compared to state-of-the-art models in low-data scenario. Our findings demonstrate the effectiveness of AulSign, with the potential to enhance accessibility and inclusivity in communication technologies for underrepresented linguistic communities.

1 Introduction

Automatic translation has seen significant advancements with Large Language Models (LLMs) [14, 37, 41]. These models, often comprising billions of parameters, excel at translating widely spoken languages. However, their performance declines significantly for low-resource languages and domain-specific text formats [7]. This limitation stems from their reliance on training data dominated by widely spoken languages, which constrains their ability to understand, represent, and translate underrepresented linguistic systems. Among low-resource languages, sign languages present significant challenges. Sign languages, such as Italian Sign Language (LIS) and American Sign Language (ASL), rely on a visual-spatial grammar system rather than spoken or written syntax. This linguistic structure, coupled with the scarcity of available training data, makes accurate translation more difficult to achieve. Furthermore, translating spoken language into sign language is under-explored in the current research. Addressing this challenge demands effective methods in data-scarce scenarios, leveraging external linguistic resources like specialized vocabularies and lexicons to produce coherent and explainable translations.

Sign Language Translation (SLT) is usually approached by separating the graphical part (computer vision or video generation) from the language part (translation) and use an intermediate language for encoding the sign language. Common intermediate representations are gloss notation, HamNoSys [31], and SignWriting [35]. Glosses provide simple written approximations using natural language labels, while HamNoSys offers a phonetic transcription of handshapes, movements, and locations in a language-independent manner. However, these systems often fail to fully capture the holistic visual-spatial structure of signed communication. SignWriting addresses these limitations by graphically encoding key features of signs (e.g., handshapes, orientations, movements, body locations) within a two-dimensional layout that mirrors the structure of signed expressions. In addition to symbolic systems, recent approaches have explored keypoint-based representations such as SMPL-X [29], which parametrically model the body, hands, and face in a continuous space, enabling direct animation of virtual avatars and bypassing the need for manual symbolic annotation. All described approaches require the availability of large amount of training data, which are not always available, especially for under-represented sign languages or in specific domains, e.g. education.

We propose Advanced Use of LLMs for Sign Language Translation (AulSign), a novel sign language translation method that can handle languages not well represented in the LLM training data.[1] Our method leverages formalized lexicons specific to a given domain, incorporating external vocabularies to enhance translation. By employing in-context learning via few-shot prompting, our method significantly reduces the need for large training corpora, while introducing a novel, transparent, and explainable translation process at each stage. Our method, which addresses both spoken-to-sign and sign-to-spoken translation tasks, comprises three core components: a Retriever, an LLM and a Sign Mapper. The Retriever module identifies and retrieves samples from a training set, which are used to instruct the LLM to map the input sentence into a pseudo-language that represents the sign language as a sequence of univocal descriptions of signs (we refer to a pseudo-language element as a canonical description). The set of samples, enriched with grammatical rules, is

* Corresponding Author. Email: gabriele.tuccio@phd.unict.it

[1] Code and supplemental material are available on https://github.com/gabrix00/AulSign

integrated into the prompt to provide the LLM with a comprehensive linguistic and structural context. For spoken-to-sign, the LLM generates the corresponding translation in pseudo-language, which is then converted into the target language by the Sign Mapper, by mapping each part of the sequence to an entry from a predefined lexicon. The process is mirrored for the inverse sign-to-spoken translation task.

We evaluate our method on two datasets covering two different spoken and sign languages: spoken Italian to and from LIS (Italian Sign Language), and spoken English to and from ASL (American Sign Language). Although our method can be applied to any structured encoding of sign language, including parametric pose models such as SMPL-X, we evaluate our approach on a SignWriting computerized specification, namely Formal SignWriting (FSW), which provides a natural way to map predicted sequences with gold sequences, therefore simplifying evaluation. FSW represents signs sequences in a linearized and standardized format suitable for computational processing, which encodes signs using sequences of symbol identifiers and positional metadata, ensuring both syntactic rigor and spatial modeling capabilities (Figure 1). This notation offers a compact, linguistically grounded, and implicitly explainable encoding of signs. It is widely adopted in education and in accessing sign language literature, and it is highly valued within the Deaf community. Our experiments demonstrate that AulSign outperforms state-of-the-art models in both spoken-to-sign and sign-to-spoken tasks, and show that LLM capabilities can be effectively leveraged for translation between spoken languages and underrepresented sign languages in a low-resource scenario. More importantly, this advancement has the potential to significantly enhance accessibility and inclusivity for the Deaf community.

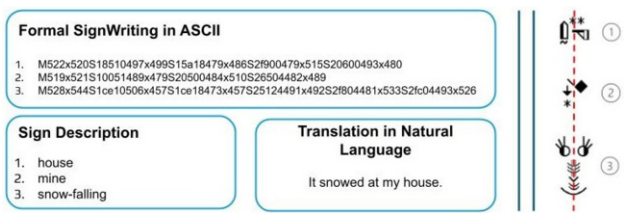

Figure 1. Example of SignWriting (right), FSW encoding and corresponding descriptions of a sign language sequence (left). FSW provides a detailed, structured representation of signs, while sign descriptions offer a more abstract and language-independent representation.

The contribution of this paper can be summarized as follows:

- We present AulSign, a novel sign language translation model able to employ LLMs for translating from and to a language that have not been seen in their training process. Designed to operate in low-resource environments, AulSign addresses data scarcity by incorporating external lexicons and structured linguistic representations into an integrated pipeline.
- Our model maps signs to a meta-lexicon, offering explainability in the translation process. This allows users to understand translation errors, detect potential misalignments, and trace inference steps.
- AulSign improves translation accuracy and usability by using detailed intermediate representations like SignWriting and SMPL-X. These help create visual content such as animated avatars, making spoken-to-sign translations more effective and accessible for Deaf users across various applications.
- We perform a comparative analysis of the performance of our method across different data availability scenarios and in a multilingual setting and show that AulSign outperforms state-of-the-art models in both spoken-to-sign and sign-to-spoken translation tasks.

The paper is structured as follows: Section 2 reviews state-of-the-art methods for sign language translation. Section 3 introduces the AulSign model, detailing its components and functionalities. Section 4 describes the experimental background and setup and presents results for both translation directions in LIS and ASL. Section 5 explores the impact of each AulSign's component, and Section 6 analyzes the findings. Finally, Section 7 presents the limitations of our work and Section 8 concludes the paper.

2 Related Works

Recent advancements in machine learning for SLT have primarily focused on two categories of models: end-to-end models, which directly map video inputs to text outputs, and representation-based models, which use intermediate representations (e.g., glosses or other meta-languages) to decompose the task into smaller, interpretable steps [9, 10]. While end-to-end models are efficient [42], gloss-based models remain popular due to their interpretability and modularity. Gloss-based multimodal models primarily address SL video-to-text translation. Inversely, SL text-to-video translation remains a more challenging task, requiring the generation of temporally coherent and linguistically accurate sign language videos. Therefore, current systems for generating sign language from text rely mainly on animation or avatar-based methods [40]. Approaches based on an intermediate representation of sign language are helpful in this case, as they bypass the complexity involved in video generation and can be concatenated with a downstream avatar animation tool. This study focuses on text-based translation systems that address both SL-to-text and text-to-SL tasks without relying on video data. State-of-the-art systems employ structured intermediate representations, including glosses [3, 39], HamNoSys [19], SignWriting and SMPL-X. The integration of HamNoSys and glosses into a text-to-avatar pipeline presents limits since HamNoSys lacks a standardized computational form suitable for direct integration into neural models, while glosses provide simple written approximations of signs using natural language labels. Gloss notation lacks the formalism to encode the phonological, spatial, and non-manual features intrinsic to sign languages, limiting their utility for synthesis or generation tasks. In contrast, representations like SignWriting or SMPL-X are explicitly designed to encode these visual-linguistic properties, making them more suitable for SL generation pipelines. Several studies have explored SignWriting's recognition and segmentation [34, 27], with most advancements focusing on languages like Arabic and Japanese [1, 25] and ontology-driven approaches [2].

Recent approaches employ transformers for translation [16, 8]. Notably, the work of Jiang et al. [16] provides key contributions to our research objectives. They propose a multilingual SLT system that translates natural language into SignWriting, setting the state-of-the-art for SignWriting-based SLT. Their multistep pipeline involves analyzing, factorizing, decoding, and evaluating SignWriting sequences to generate natural language text. The system employs sequences of FSW, which are decomposed into base units (symbols) and supplementary information (e.g., positional numbers). These elements are encoded as variables processed by a factorized model [20, 12]. For spoken-to-sign tasks, symbol tokens are decoded using Beam Search, while positional factors are predicted through regression. Additionally, dictionary data (e.g., Dictionary Puddles)[2] are integrated to enhance model performance. Despite

[2] https://www.signbank.org/signpuddle/

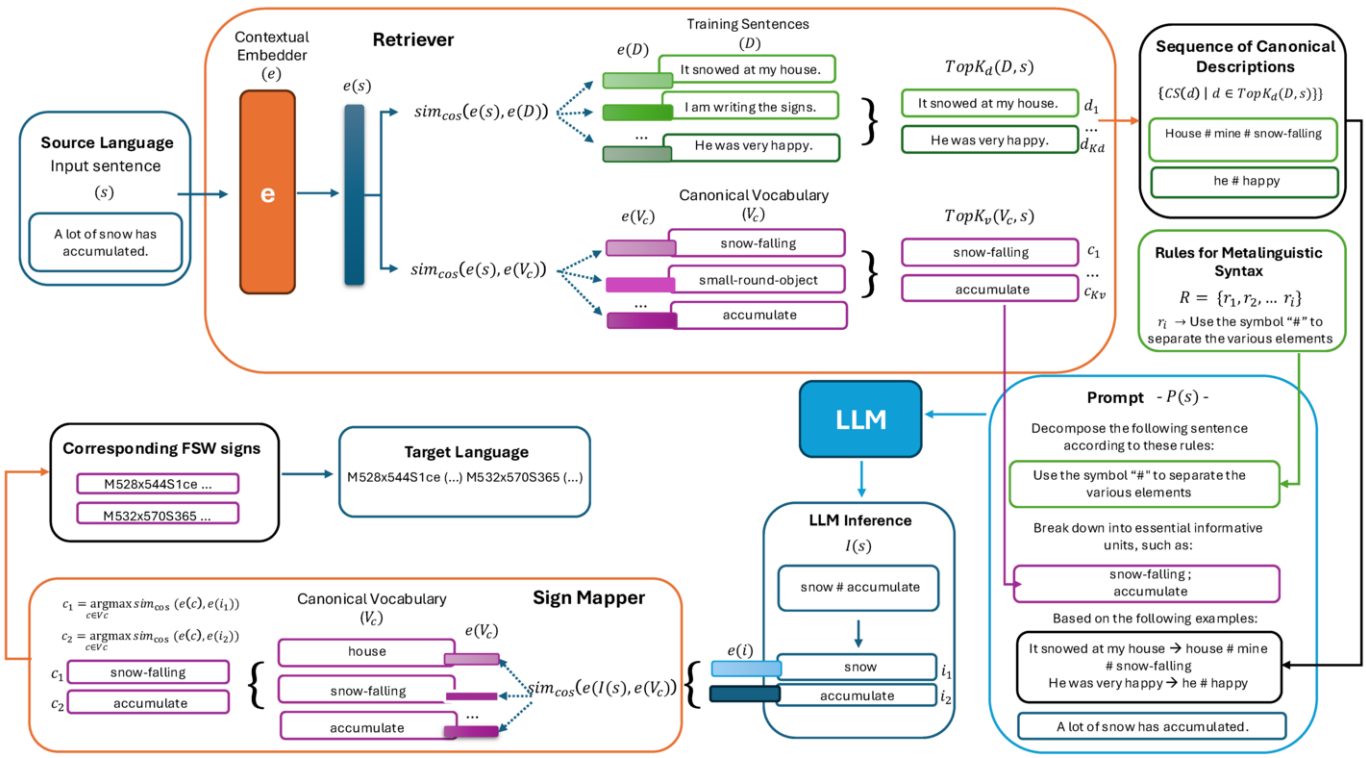

Figure 2. Overview of the spoken-to-sign AulSign pipeline. We detail our method using the FSW notation as a reference.

these advancements, the model shows some limitations. For example, finger-spelling requires specialized handling, and traditional text classification metrics may be insufficient for evaluating SLT due to its visual-spatial characteristics. Recent advancements have leveraged LLMs to enhance SLT, particularly in video-to-sign language tasks [13, 15, 23, 38, 17]. These methods have been explored with and without intermediate representations like gloss notation [24, 18]. Furthermore, LLMs have demonstrated potential in translating between text and glosses [11, 21]. However, to the best of our knowledge, no research has directly addressed the task of Sign Language translation using generative pretrained LLMs from natural language to encoded language representations, such as SignWriting. This aspect poses challenges since LLMs have not seen sign language representations during their training phase.

3 AulSign: A novel LLM-based method for Sign Language Translation

Our method comprises three main components: a Retriever, an LLM, and a Sign Mapper. The Retriever identifies samples from the training set[3] The training set is pre-processed offline to convert signs into canonical descriptions, which serve as pseudo-language expressions capturing the meaning of each sign. Compared to glosses, canonical descriptions are longer, more expressive, and designed to be unique for each specific meaning, allowing a one-to-one correspondence between a description and a sign. This design avoids the ambiguity arising from polysemy and enhances semantic precision. By maintaining explicit and interpretable mappings, our method not only improves explainability but also allows adaptation to different structured sign encodings, such as SignWriting, HamNoSys, and SMPL-X.

Each sample of the pre-processed training set contains the spoken text and a corresponding decomposition into a sequence of canonical descriptions. Samples that are returned by the Retriever are used for prompt generation. This task combines the retrieved sample sentences, paired with their corresponding decompositions, with samples from a structured vocabulary and a set of predefined grammar rules to employ a structured prompt, which enables the LLM to perform few-shot inference on the input sentence. The Sign Mapper is employed only in spoken-to-sign translation and converts the sequence of canonical descriptions generated by the LLM into a sequence of signs. The whole process is described in Figure 1. Next we detail the various parts of our pipeline in the spoken-to-sign direction. The inverse process is similar, with small adjustments, which are discussed at the end of the section.

Our method considers a training set D of spoken sentences associated with their sign's encoding counterpart, i.e. sequences of signs, and a vocabulary of signs V_s, where each sign is associated with one or more descriptions in natural language. We pre-process both D and V_s to translate the encoded sign sequences into sequences of canonical descriptions[4]. First, we define an equivalence operator[5] $\equiv$ between signs and merge equivalent signs into one single entry, choosing the most frequent sign as a representative. Then, we construct a set V_c of unique and non-ambiguous canonical descriptions, by choosing among all descriptions associated to a sign the most frequent ones and combining them into a single string. Finally we substitute the sequences of signs associated to sentences in D with corresponding sequences of canonical descriptions by probing, for

[3] we refer to the set of available sentences with corresponding translation, although our method do not require explicit training

[4] for the Italian LaCAM CNR-ISTC dataset we do not need this step since in the corpus all signs have been manually associated to canonical descriptions

[5] in our implementation signs are considered equivalent if they contain the same set of symbols with associated orientation and rotation; more complex notions that also consider the spatial position can be employed to improve results

each sign of the sequence, an equivalent sign from V_s and taking the corresponding canonical description. If an equivalent sign exists in V_s, it is unique because all equivalent signs have been previously merged in V_s. On the other hand, if an equivalent sign is not found, a special "<unk>" description is considered. Given a spoken sentence $d \in D$, we denote as $CS(d)$ its associated sequence of canonical descriptions[6].

The Retriever identifies the top-K sentences from the training set D that are most semantically similar to the given input sentence s. This process ensures that the subsequent prompt generation operates with contextually relevant examples. To improve results, the Retriever also identifies a set of canonical descriptions from V_c that are similar to the input sentence, to be fed to the LLM as samples of canonical descriptions. Both retrieval steps employ a contextual embedder $e(\cdot)$, trained for semantic similarity, to encode both the input sentence and the candidate sentences into high-dimensional representations. The similarity between s and each candidate is computed by cosine similarity. The system selects the top-K most semantically similar entries, which we denote with the sets $\mathit{TopK}_d(D, s)$ and $\mathit{TopK}_v(V_c, s)$, respectively.

The prompt generation step develops an input-sentence-customized prompt, leveraging on the few-shot learning abilities of LLMs. We design the prompt by combining grammar rules ($R = \{r_1, r_3, \dots r_i\}$) with illustrative examples – retrieved sentences with their canonical descriptions – and the input sentence. The retrieved canonical descriptions from the Retriever are also included in the prompt as examples of canonical descriptions that the LLM can use to break down the input sentence. The complete prompt structure is shown in Figure 1.

The LLM generates a sequence $I(s)$ of strings in a form that mimics canonical descriptions in V_c, which feeds the Sign Mapper. To match the generated pseudo-canonical descriptions to entries in V_c we again use semantic similarity. We employ the same contextual embedder $e(\cdot)$ used for retrieval and, for each pseudo-canonical $i_l \in I(s)$, we extract: $c_l = \arg\max_{c \in V_c} \mathrm{sim}_{\cos}(e(c), e(i_l))$. Finally, the model retrieves the corresponding encoded signs associated with the selected canonical descriptions to generate the final translation. The output of AulSign can be provided to an avatar animation system, which can be implemented rule based, i.e. associating real poses and movement to SignWriting glyphs, or by interpolating between key SMPL-X poses [4].

For the sign-to-spoken task, the input sentence in sign language notation is first converted into a sequence of canonical descriptions in the same way as for the training set pre-processing. Signs of the input sequence are probed from V_s using the equivalence operator $\equiv$ and the corresponding canonical descriptions are taken. Similar sequences of canonical descriptions are then extracted from D and used with their speech counterpart to instruct the LLM, i.e. to build the prompt with few-shot samples. The prompt is built in the same way as for speech-to-sign, except that the examples for the break down are not given, since they are not necessary. The output of the LLM is then returned as the resulting spoken sentence.

4 Experiments and Results

To assess the effectiveness of AulSign, we conduct experiments on two benchmark datasets: the English SignBank+ [26] for ASL and the LaCAM CNR-ISTC Italian dataset for LIS [5]. SignBank+ is a multilingual corpus with 254,002 distinct elements, 76 sign language encodings, and 153 "puddles", systems that categorize signs by language or dialect. The dataset is organized into three subcorpora: Manually (cleaned via manual review), GPT-3.5 (cleaned using GPT-3.5 with a few-shot learning paradigm [6]), and Bible (aligned biblical texts). To ensure consistency and reduce noise, we focus exclusively on the English subset for our experiments. The English subcorpus of SignBank+ comprises 43,705 annotated items spanning nine variants of English Sign Language, including 19,304 unique signs and 13,631 sentences. We use the unique-sign English SignBank+ subcorpus to build V_c (cf. Sect. 3) and split the sentence-based subcorpus into training set D (cf. Sect. 3) and test set. We experiment with three different training set configurations (referred to as I, II, III) to assess the method's robustness across varying data conditions. The first configuration considers the entire training set, which counts 13,275 sentences. Note that not all signs in the sentences are covered by our vocabulary (extracted by single sign sentences), therefore our method, which requires a complete vocabulary, is at a disadvantage in this setting[7]. The second configuration filters out sentences containing out-of-vocabulary signs, resulting in a more consistent dataset of 2,301 sentences. The third configuration simulates a low-resource scenario by randomly sampling 115 sentences from the training set of the second configuration. Notably, although the number of training samples varies across configurations, the vocabulary V_c remains fixed as it serves as a connection between canonical descriptions and their corresponding entries in the encoded sign representation. For comparisons, we train the state-of-the-art model from Jiang et al. [16] using three parallel configurations (i.e. I, II, and III). Each model is trained on the same set of training sentences as the corresponding AulSign configurations and includes the complete vocabulary (i.e. V_c) in each configuration training process. Following Jiang et al., we configure the set of hyperparameters with a learning rate of 0.0001, a drop-out of 0.50, a batch size of 64[8] and a learning-rate-factor of 0.70. We evaluate both approaches on a test set of 356 sentences.

The LaCAM CNR-ISTC corpus [5] is an Italian dataset based on SignPuddle[9] for LIS. It contains 2,149 annotated items, including 1,974 signs and 782 sentences. Each sign is associated with a handmade canonical description and a FSW representation. We use signs to populate the vocabulary V_c and split the dataset into training and test sets of 547 and 235 sentences, respectively. Due to the limited data available in the Italian LIS dataset, we do not define three separate configurations as in the ASL task.

To retrieve relevant examples for the AulSign Retriever module, we employ the Lee et al. [22] and the Reimers et al. [32] models for ASL and LIS, respectively, as embedding models. We set the number of top retrieved sentences K_d to 20 and the number retrieved canonical descriptions K_v to 100.

For prompt generation (cf. Sect. 3), we define two distinct sets of rules to structure the metalinguistic syntax settings. These rules describe each sign based on its canonical description and are supplemented with illustrative examples of sentences and their associated canonical descriptions to ensure clarity and alignment between linguistic concepts and their representations. We define a total of seven rules for both English and Italian[10]. For all experiments, we employ GPT-3.5 as an LLM.[11]

Section 4.1 details the overall performance of the AulSign method

[6] further details and examples are provided in the supplemental material

[7] we identify the extraction of vocabulary from sentences as an interesting research direction

[8] in place of 32 used by Jiang et al. to improve efficiency

[9] https://www.signbank.org/signpuddle/

[10] for an overview of the grammar rules employed, we refer to https://github.com/gabrix00/AulSign

[11] https://platform.openai.com/docs/models/gpt-3-5-turbo

for LIS and ASL in executing the spoken-to-sign translation task. Section 4.2 presents the results for the inverse task, i.e., sign-to-spoken translation. To provide a deeper understanding of our results, we performed both an error analysis and a cost analysis, the details of which are presented in the supplemental materials.

4.1 Spoken-to-sign translation

Table 1 presents the overall performance of the AulSign method on the English SignBank+ dataset in terms of F1-score, BLEU [28], ChrF2 [30], and Mean Absolute Error (MAE). We consider the F1-score as an appropriate metric to evaluate predictions at the symbol (with associated rotation and orientation) level, since the order of symbols within a sign is not semantically relevant[12]. We compute the F1 score on a sentence-by-sentence basis by considering all signs collectively and calculating the harmonic mean of precision and recall. We then report the average across the entire dataset. This approach accounts for the challenge of aligning gold and predicted signs within a sentence, especially when their lengths or ordering differ. However, symbol-level evaluation alone is insufficient for FSW, due to the semantically relevant order across signs. To address this limitation, and following Jiang et al. [16], we also employ the popular translation metrics BLEU [28] and ChrF2 [30], which capture statistics at both word and character levels. Additionally, we assess positional values (x and y) using MAE to quantify the discrepancy between predicted and ground-truth values. As the symbol order within a sign is not semantically significant, both gold standard and predicted sequences are alphabetically sorted before evaluation.

Table 1. Overall spoken-to-sign ASL translation models performance on the English SignBank+ dataset, in terms of F1 score, BLEU, ChrF2, and MAE. We test three AulSign configurations: (I) data-rich (13,275 sentences), (II) filtered (2,301 sentences), and (III) low-resource (115 sentences). $|V_c|$ refers to the totality of vocabulary entries. We compare the results to the baseline model (Jiang et al. [16]) under the same conditions, with bold values indicating the best result per configuration.

Setting	Model	F1	BLEU	ChrF2	MAE X	MAE Y	Training size
I	Jiang et al.	**0.45**	**29.26**	**57.82**	**23.77**	**27.96**	13,275 + $\|V_c\|$
I	AulSign	0.42	25.40	56.44	25.02	29.66	
II	Jiang et al.	0.40	22.68	50.65	**23.78**	**27.80**	2,301 + $\|V_c\|$
II	AulSign	**0.41**	**23.96**	**56.06**	25.42	30.31	
III	Jiang et al.	0.27	10.94	39.17	**23.81**	**27.82**	115 + $\|V_c\|$
III	AulSign	**0.37** (±0.006)	**18.79** (±0.665)	**53.91** (±0.153)	25.52 (±0.138)	30.95 (±0.188)	

As shown in Table 1, AulSign consistently outperforms the Jiang et al. baseline in data-scarce scenarios. In this scenario, the model achieves an F1 score of 0.37, a 10-point improvement over Jiang et al. III configuration (i.e. 0.27). This trend is mirrored in BLEU and ChrF2 scores, where the third AulSign configuration attains 18.79 and 53.91, compared to 10.94 and 39.17 for the baseline. These results underscore the AulSign's ability to handle low-resource settings. In data-rich settings (I and II), AulSign demonstrates comparable performance to Jiang et al., even when confronted with noisy data. AulSign I achieves an F1 score of 0.42, slightly lower than Jiang et al. I (i.e. 0.45), while maintaining competitive BLEU (i.e. 25.40 vs. 29.26) and ChrF2 scores (i.e. 56.44 vs. 57.82). Our assessment affirms AulSign's robustness across different data conditions. Regarding MAE, Jiang et al.'s models generally achieve lower positional errors, with minimum values (in configuration II) of 23.77 and 27.80 for X and Y coordinates, respectively. AulSign's MAE values are slightly higher (25.42 and 30.31 for the same configuration), reflecting a trade-off between precise factor prediction and overall symbol recognition quality. This discrepancy can be attributed to the architectural focus of Jiang et al.'s models on positional accuracy, whereas AulSign employs a general-purpose translation framework. Nevertheless, AulSign achieves competitive overall performance, highlighting its efficacy in handling both sequence-level and positional challenges in SignWriting-based translation. To better illustrate the performance trend as the amount of training data varies, we report in Figure 3 the F1-score column of Table 1. AulSign shows consistent performance across training set sizes, contrasting with Jiang et al.'s model, which exhibits significant degradation in low-resource conditions. To assess the robustness of our method even with a small number of samples, in the low-resource configuration (III) we compute the average and standard deviation of 10 runs, each of them obtained with a different subset of 115 sentences. In the other configurations we use the whole set of available sentences (with and without "<unk>"), therefore we cannot perform this test. Results show a standard deviation of less than 1%.

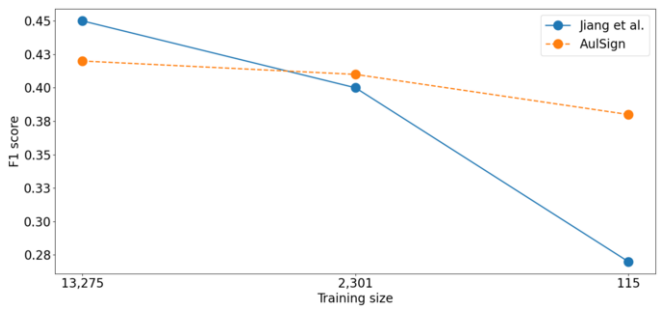

Figure 3. Performance trend of AulSign and the state-of-the-art model across different training set sizes, evaluated in terms of F1-score (first column Table 1) for the ASL spoken-to-sign translation task.

Table 2 summarizes the performance of AulSign on the Italian LaCAM CNR-ISTC corpus, evaluated using F1-score, BLEU, ChrF2, and MAE for positional coordinates. Since the Italian setting represents a low-resource scenario with limited data availability, we cannot compute standard deviations or report results for additional data configurations, as further subdivisions would be uninformative. AulSign achieves substantial improvements across all metrics compared to Jiang et al. Notably, it attains an F1-score of 0.63, representing a 13-point increase over the baseline (0.50), demonstrating superior capability in sign classification and identification. For sequence-level metrics, AulSign outperforms Jiang et al. with BLEU and ChrF2 scores of 37.71 and 57.54, respectively, compared to the baseline's 16.40 and 45.18. Additionally, AulSign demonstrates superior positional accuracy, with MAE values of 21.22 (X) and 33.22 (Y), compared to 23.82 and 37.44 for Jiang et al. This indicates AulSign's ability to precisely model even the spatial characteristics of signs when a comprehensive vocabulary is available.

Table 2. Models overall performance in spoken-to-sign Italian-to-LIS translation using FSW-encoded sequences on the Italian LaCAM CNR-ISTC corpus.

Model	F1	BLEU	ChrF2	MAE X	MAE Y
Jiang et al.	0.50	16.40	45.18	23.82	37.44
AulSign	**0.63**	**37.71**	**57.54**	**21.22**	**33.22**

4.2 Sign-to-spoken translation

Tables 3 and 4 show the results on sign-to-spoken translation for ASL-to-English and LIS-to-Italian, using FSW-encoded sign language sequences. As for spoken-to-sign, we evaluate three AulSign model configurations (I, II, III) for ASL and one for LIS in terms of

[12] for further information about the structure of FSW we refer to [16]

BLEU and ChrF2. We do not compute F1 and MAE since the former is not adequate to assess spoken natural language sentences and the latter has no meaning in this context. For a comprehensive overview of model configurations and experimental setup, we refer to the beginning of this section.

As shown in Table 3 and Table 4, AulSign consistently outperforms the baseline, achieving state-of-the-art results for both ASL and LIS. In a low-resource setting for the ASL, AulSign significantly outperforms the Jiang et al. model, yielding a BLEU score of 26.59 and a ChrF2 score of 40.76, compared to the baseline's 2.20 and 18.90, respectively. Similarly, AulSign achieves a BLEU score of 17.95 and a ChrF2 score of 50.42, significantly outperforming the baseline scores of 6.50 and 29.40 in the low-resource LIS setting. In high-resource settings, we evaluate two configurations of AulSign (i.e. I and II). Both configurations show superior performance compared to the Jiang et al. model. Specifically, the first configuration of AulSign achieves a BLEU score of 23.38 and a ChrF2 score of 39.21, exceeding the baseline scores of 18.40 and 34.40. Notably, the second configuration, trained on a reduced dataset, significantly outperforms the baseline, with a BLEU score of 24.75 and a ChrF2 score of 40.26, compared to the baseline's 8.40 and 22.50, respectively. In low-resource scenarios, the low performances of Jiang et al. are expected since there are not sufficient data for training, considering the complexity of natural language. In any case, our model can leverage the language processing abilities of pretrained LLMs, therefore producing higher-quality translation. As for the spoken-to-sign task (cf. Sect. 4.1), we also computed an average on 10 different runs to assess the robustness of our method for the ASL low-resource scenario (configuration III, not reported). Performance resulted similar to those of Table 3, with a standard deviation of less than 1%.

Table 3. Overall sign-to-spoken ASL translation models performance on the English SignBank+ dataset, in terms of BLEU, and ChrF2. We test three AulSign configurations: (I) data-rich (13,275 sentences), (II) filtered (2,301 sentences), and (III) low-resource (115 sentences). $|V_c|$ refers to the totality of vocabulary entries. We compare the results to the baseline model (Jiang et al. [16]) under the same conditions. Bold values indicate the best result per configuration.

Setting	Model	BLEU	ChrF2	Training Size
I	Jiang et al.	18.40	34.40	13,275 + $\|V_c\|$
I	AulSign	**23.38**	**39.21**	
II	Jiang et al.	8.40	22.50	2,301 + $\|V_c\|$
II	AulSign	**24.75**	**40.26**	
III	Jiang et al.	2.20	18.90	115 + $\|V_c\|$
III	AulSign	**26.59**	**40.76**	

Table 4. Overall performance in sign-to-spoken LIS translation using FSW-encoded sequences on the Italian LaCAM CNR-ISTC dataset, evaluated in terms of BLEU and ChrF2.

Model	BLEU	ChrF2
Jiang et al.	6.50	29.40
AulSign	**17.95**	**50.42**

5 Ablation Study

We assess the contribution of the AulSign framework's individual components by analyzing the impact of its three core modules: the Retriever, the LLM, and the Sign Mapper (cf. Sect. 3). The ablations are conducted on both spoken-to-sign and sign-to-spoken tasks for ASL[13]. To assess AulSign robustness, we experiment models performance on a low-resource setting, i.e., on the 115 items of configuration III (cf. Sect. 4.2), reporting the mean over 10 different runs with the corresponding standard deviation.

Table 5 examines how variations in the embedding model affect performance, highlighting the roles of the Retriever and the Sign Mapper. The embedder plays a critical role in two stages: (i) during the Retriever phase, where it selects the most semantically similar sentences from the training set to construct the LLM prompt, and (ii) in the Sign Mapper module, where it aligns the pseudo-canonical descriptions generated by the LLM with entries from the canonical vocabulary. We report the performance of three well-known embedding models, i.e., MPnet [33], MiniLM [33] and Mxbai [22] in terms of F1-score, BLEU, and ChrF2. The MPNet model, based on a masked and permuted pre-training architecture, shows strong performance in capturing semantic similarity and provides high accuracy across standard benchmarks. The MiniLM model, while lighter and faster due to its distilled transformer architecture, offers a good trade-off between performance and efficiency, making it suitable for real-time or resource-constrained environments. Finally, the Mxbai model, a large-scale embedder optimized through instruction tuning, outperforms the others in terms of generalization and retrieval tasks, demonstrating particularly robust results in multi-domain and cross-task scenarios. As shown in Table 5, all models exhibit comparable performance, with low standard deviation (i.e., less than 2%) across both text-to-sign and sign-to-text tasks. Specifically, MxBai achieves slightly higher scores in the text-to-sign direction, while MPNet performs marginally better in the sign-to-text task. However, these differences are minimal, suggesting AulSign is robust to changing the embedder.

Table 6 analyzes the impact of the LLM module by comparing three different AulSign's configurations based on distinct LLMs. Specifically, we evaluate the performance of LLaMA-3 70B[14], GPT-4.1-nano[15], and GPT-3.5 to assess the robustness of the proposed method across varying model architectures and capacities. LLaMA-3 70B is a state-of-the-art open-weight model developed by Meta, offering strong instruction-following capabilities. GPT-4.1-nano is an efficiency-oriented variant of the GPT-4 family, optimized for low-latency deployment with reduced computational cost. GPT-3.5, a widely adopted predecessor to GPT-4, provides a strong performance baseline with balanced accuracy and efficiency. As shown in Table 6, GPT-3.5 outperforms the other models, achieving the highest scores in terms of F1, BLEU, and ChrF2 across both translation directions. However, the performance gap relative to the other models is modest, with differences limited to a few percentage points. Moreover, the results are consistent and robust, as evidenced by the low standard deviation observed across all configurations.

6 Discussion

This study evaluates AulSign for SLT, particularly in low-resource scenarios, using SignWriting as an intermediate representation. AulSign achieves state-of-the-art results for both ASL and LIS in spoken-to-sign and sign-to-spoken tasks, consistently outperforming Jiang et al. [16] across different data configurations. Notably, AulSign excels in data-scarce environments, demonstrating strong generalization and maintaining performance even with reduced training data.

AulSign is shown to effectively produce linguistically accurate and spatially coherent sign language translations, even with limited

[13] we report ablation studies for LIS in the supplementary materials, as they have similar performance to those of ASL

[14] https://ollama.com/library/llama3.3:70b-instruct-fp16

[15] https://platform.openai.com/docs/models/gpt-4.1-nano

Table 5. Spoken-to-Sign and Sign-to-Spoken ASL ablation study of AulSign focusing on the embedder.We test three different embedder models (i.e. MPnet, MiniLM and Mxbai) on the English SignBank+ dataset in a low-resource scenario (i.e. 115 items). For each score, we report the mean of 10 runs with the standard deviation.

Task	Embedder	F1	BLEU	ChrF2	MAE X	MAE Y
Text-to-Sign	MPnet	0.34 (± 0.005)	14.71 (± 0.444)	54.38 (± 0.174)	25.13 (± 0.111)	31.25 (± 0.191)
Text-to-Sign	MiniLM	0.34 (± 0.00)	15.16 (± 0.425)	54.73 (± 0.248)	25.71 (± 0.137)	31.05 (± 0.122)
Text-to-Sign	Mxbai	0.34 (± 0.004)	15.43 (± 0.316)	54.92 (± 0.213)	25.36 (± 0.126)	31.36 (± 0.259)
Sign-to-Text	MPnet	-	19.92 (± 0.638)	39.91 (± 0.230)	-	-
Sign-to-Text	MiniLM	-	19.05 (± 1.590)	39.55 (± 1.063)	-	-
Sign-to-Text	Mxbai	-	18.89 (± 1.909)	39.13 (± 1.046)	-	-

Table 6. Spoken-to-Sign and Sign-to-Spoken ASL ablation study focusing on the LLM.We test AulSign performance at the variation of three different LLMs (i.e. LlaMa-70B, GPT-3.5 and GPT-4.1-nano) on the English SignBank+ dataset in a low-resource scenario (i.e. 115 items). For each score, we report the mean of 10 runs with the standard deviation.

Task	LLM	F1	BLEU	ChrF2	MAE X	MAE Y
Text-to-Sign	LlaMa-70B	0.34 (± 0.004)	15.43 (± 0.316)	54.92 (± 0.213)	25.36 (± 0.126)	31.36 (± 0.259)
Text-to-Sign	GPT-3.5	0.37 (± 0.006)	19.73 (± 0.735)	55.07 (± 0.159)	25.52 (± 0.138)	30.95 (± 0.188)
Text-to-Sign	GPT-4.1-nano	0.35 (± 0.003)	17.66 (± 0.145)	55.52 (± 0.196)	25.18 (± 0.077)	30.77 (± 0.147)
Sign-to-Text	LlaMa-70B	-	18.89 (± 1.909)	39.14 (± 1.046)	-	-
Sign-to-Text	GPT-3.5	-	24.22 (± 1.139)	40.04 (± 0.750)	-	-
Sign-to-Text	GPT-4.1-nano	-	15.29 (± 0.718)	32.38 (± 0.652)	-	-

data. Its strong performance in sign-to-spoken tasks is due to its use of context augmentation and domain-specific vocabulary mapping. AulSign adapts well to varying data sizes, maintaining consistent results, unlike the Jiang et al. model, which struggles with smaller datasets. While AulSign is slightly behind the baseline in some data-rich scenarios, its adaptability is superior, especially with smaller or higher-quality datasets.

Across tasks, AulSign demonstrates balanced performance. In LIS translation, AulSign consistently outperforms Jiang et al., benefiting from the highest quality level of the Italian LaCAM CNR-ISTC dataset. AulSign's approach leverages a predefined, domain-specific vocabulary, which plays a central role in improving token alignment and translation accuracy. This strategy ensures precise mappings between signs and their representations, particularly in structured intermediate formats like SignWriting. However, signs not present in the vocabulary are treated as unknown, which may affect the model's ability to handle dynamic linguistic contexts or rare signs. Expanding the vocabulary to cover a broader range of linguistic variations and incorporating mechanisms for inferring or adapting to unseen signs would further enhance the model's capacity to address diverse translation challenges. This is particularly relevant in high-resource settings, where the model's performance is highly sensitive to vocabulary size and coverage, leading to comparable but slightly lower performances in the full-data spoken-to-sign ASL configuration (scenario I). Finally, AulSign's modular architecture ensures transparency at each stage of the translation pipeline. The retrieval module, for instance, suitably select in-context samples, while the prompt generation phase explicitly aligns input sentences with grammatical rules. This design allows for targeted error analysis, such as identifying misalignments between FSW symbols and canonical descriptions, which is infeasible in end-to-end SLT models. By prioritizing explainability, AulSign bridges the gap between high performance and user trust, offering a reliable tool for researchers and end-users.

7 Limitations

While AulSign demonstrates promising results, certain limitations affect its generalizability and broader applicability. The model relies on a vocabulary of canonical descriptions to map signs to natural language, ensuring consistency and control over translation quality. However, this dependency introduces a degree of rigidity, as it relies on a specialized vocabulary in which a one-to-one correspondence between signs and their descriptions must be explicitly defined. Furthermore, although AulSign is designed to perform effectively in low-resource settings, its translation quality remains contingent on the availability and consistency of training data. Incomplete or inconsistently annotated datasets may adversely impact accuracy, particularly in capturing the spatial positioning of signs. At present, this study employs SignWriting as the primary intermediate representation due to its structured and expressive nature. Nonetheless, the approach is inherently extensible to alternative notational systems, such as HamNoSys, pose representations, such as SMPL-X, or specialized lexical glossaries, thereby offering potential avenues for enhanced adaptability and integration within automatic sign language translation frameworks. Finally, the current reliance of the Sign Mapper on rigid vocabulary matching could be mitigated through grammar-constrained decoding methods [36], which would allow the generation process to directly enforce structural and syntactic constraints, thereby reducing dependence on explicit one-to-one mappings.

8 Conclusion

AulSign is a novel framework for sign language translation that leverages retrieval-augmented generative models for enabling translation to previously unseen languages. It achieves state-of-the-art performance in spoken-to-sign and sign-to-spoken translation for both English ASL and Italian LIS, excelling in low-resource scenarios. By using structured vocabularies and a modular, explainable architecture, AulSign enhances translation accuracy, interpretability, and error analysis. Future work aims to integrate multimodal data, increase adaptability, and extend support to additional sign languages.

9 Acknowledgment

We acknowledge financial support from the Next Generation EU Program with the Future Artificial Intelligence Research (FAIR) project, code PE00000013, CUP 53C22003630006, and by the PNRR project Learning for All (L4ALL) funded by the Italian MIMIT (number: F/310072/01-05/X56). We thank the members of the LaCAM Laboratory, especially Chiara Bonsignori, Alessio Di Renzo, Gabriele Gianfreda, Luca Lamano, Tommaso Lucioli, Barbara Pennacchi, and the LaCAM supervisor, Olga Capirci.

References

[1] A. M. Almasoud and H. S. Al-Khalifa. A proposed semantic machine translation system for translating arabic text to arabic sign language. In *Proceedings of the Second Kuwait Conference on e-Services and e-Systems*, pages 1–6, 2011.

[2] A. M. Almasoud and H. S. Al-Khalifa. Semsignwriting: A proposed semantic system for arabic text-to-signwriting translation. 2012.

[3] G. Angelova, E. Avramidis, and S. Möller. Using neural machine translation methods for sign language translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 273–284, 2022.

[4] M. Aziz and A. Othman. Evolution and trends in sign language avatar systems: Unveiling a 40-year journey via systematic review. *Multimodal Technologies and Interaction*, 7(10):97, 2023.

[5] C. Bonsignori, A. Di Renzo, G. Gianfreda, T. Lucioli, B. Pennacchi, L. Lamano, and O. Capirci. A visually annotated multimodal corpus of lis, international workshop multimodal language: theoretical perspectives and research methods. In *Proceedings of the 2019 international symposium on signal processing systems*, 2025.

[6] T. B. Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[7] J. S. Daniel and A. Pal. Impact of non-standard unicode characters on security and comprehension in large language models. *arXiv preprint arXiv:2405.14490*, 2024.

[8] F. de Almeida Freitas, S. M. Peres, O. de Paula Albuquerque, and M. Fantinato. Leveraging sign language processing with formal signwriting and deep learning architectures. In *Brazilian Conference on Intelligent Systems*, pages 299–314. Springer, 2023.

[9] M. De Coster, D. Shterionov, M. Van Herreweghe, and J. Dambre. Machine translation from signed to spoken languages: State of the art and challenges. *Universal Access in the Information Society*, 23(3):1305–1331, 2024.

[10] U. Farooq, M. S. M. Rahim, N. Sabir, A. Hussain, and A. Abid. Advances in machine translation for sign language: approaches, limitations, and challenges. *Neural Computing and Applications*, 33(21): 14357–14399, 2021.

[11] P. Fayyazsanavi, A. Anastasopoulos, and J. Košecká. Gloss2text: Sign language gloss translation using llms and semantically aware label smoothing. *arXiv preprint arXiv:2407.01394*, 2024.

[12] M. García-Martínez, L. Barrault, and F. Bougares. Factored neural machine translation architectures. In *Proceedings of the 13th International Conference on Spoken Language Translation*, 2016.

[13] J. Gong, L. G. Foo, Y. He, H. Rahmani, and J. Liu. Llms are good sign language translators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18362–18372, 2024.

[14] M. U. Hadi, Q. Al Tashi, A. Shah, R. Qureshi, A. Muneer, M. Irfan, A. Zafar, M. B. Shaikh, N. Akhtar, J. Wu, et al. Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects. *Authorea Preprints*, 2024.

[15] E. J. Hwang, S. Cho, J. Lee, and J. C. Park. An efficient sign language translation using spatial configuration and motion dynamics with llms. *arXiv preprint arXiv:2408.10593*, 2024.

[16] Z. Jiang, A. Moryossef, M. Müller, and S. Ebling. Machine translation between spoken languages and signed languages represented in signwriting. *arXiv preprint arXiv:2210.05404*, 2022.

[17] Z. Jiang, G. Sant, A. Moryossef, M. Müller, R. Sennrich, and S. Ebling. Signclip: Connecting text and sign language by contrastive learning. *arXiv preprint arXiv:2407.01264*, 2024.

[18] L. E. Johnson and S. Rashad. An innovative system for real-time translation from american sign language (asl) to spoken english using a large language model (llm). In *2024 IEEE 15th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pages 605–611. IEEE, 2024.

[19] Z. Kang. Spoken language to sign language translation system based on hamnosys. In *Proceedings of the 2019 international symposium on signal processing systems*, pages 159–164, 2019.

[20] P. Koehn and H. Hoang. Factored translation models. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 868–876. Association for Computational Linguistics, 2007.

[21] H. Lee, J.-H. Kim, E. J. Hwang, J. Kim, and J. C. Park. Leveraging large language models with vocabulary sharing for sign language translation. In *2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*, pages 1–5. IEEE, 2023.

[22] S. Lee, A. Shakir, D. Koenig, and J. Lipp. Open source strikes bread - new fluffy embeddings model, 2024. URL https://www.mixedbread.ai/blog/mxbai-embed-large-v1.

[23] J. Lim, I. Sa, B. MacDonald, and H. S. Ahn. A sign language recognition system with pepper, lightweight-transformer, and llm. *arXiv preprint arXiv:2309.16898*, 2023.

[24] Y. Liu, W. Zhang, S. Ren, C. Huang, J. Yu, and L. Xu. Scope: Sign language contextual processing with embedding from llms. *arXiv preprint arXiv:2409.01073*, 2024.

[25] T. Matsumoto, M. Kato, and T. Ikeda. Jspad: A sign language writing tool using signwriting. In *proceedings of the 3rd international universal communication symposium*, pages 363–367, 2009.

[26] A. Moryossef and Z. Jiang. Signbank+: Multilingual sign language translation dataset. *arXiv preprint arXiv:2309.11566*, 2023.

[27] A. Moryossef, Z. Jiang, M. Müller, S. Ebling, and Y. Goldberg. Linguistically motivated sign language segmentation. *arXiv preprint arXiv:2310.13960*, 2023.

[28] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[29] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019.

[30] M. Popović. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the tenth workshop on statistical machine translation*, pages 392–395, 2015.

[31] S. Prillwitz and H. Zienert. Hamburg notation system for sign language: Development of a sign writing with computer application. In *Current trends in European Sign Language Research. Proceedings of the 3rd European Congress on Sign Language Research*, pages 355–379, 1990.

[32] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL http://arxiv.org/abs/1908.10084.

[33] N. Reimers and I. Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2020. URL https://arxiv.org/abs/2004.09813.

[34] A. F. Sevilla, A. D. Esteban, and J. M. Lahoz-Bengoechea. Automatic signwriting recognition: Combining machine learning and expert knowledge to solve a novel problem. *IEEE Access*, 11:13211–13222, 2023.

[35] V. Sutton. *Lessons in SignWriting*. SignWriting Press, 2022.

[36] G. Tuccio, L. Bulla, M. Madonia, A. Gangemi, and M. Mongiovì. GRAMMAR-LLM: Grammar-constrained natural language generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3412–3422, 2025.

[37] L. Wang, C. Lyu, T. Ji, Z. Zhang, D. Yu, S. Shi, and Z. Tu. Document-level machine translation with large language models. *arXiv preprint arXiv:2304.02210*, 2023.

[38] R. Wong, N. C. Camgoz, and R. Bowden. Sign2gpt: Leveraging large language models for gloss-free sign language translation. *arXiv preprint arXiv:2405.04164*, 2024.

[39] K. Yin and J. Read. Better sign language translation with stmc-transformer. *arXiv preprint arXiv:2004.00588*, 2020.

[40] Z. Yu, S. Huang, Y. Cheng, and T. Birdal. Signavatars: A large-scale 3d sign language holistic motion dataset and benchmark. In *European Conference on Computer Vision*, pages 1–19. Springer, 2025.

[41] B. Zhang, B. Haddow, and A. Birch. Prompting large language model for machine translation: A case study. In *International Conference on Machine Learning*, pages 41092–41110. PMLR, 2023.

[42] B. Zhou, Z. Chen, A. Clapés, J. Wan, Y. Liang, S. Escalera, Z. Lei, and D. 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.