Pipeline Signed Japanese Translation Focusing on a Post-positional Particle Complement and Conjugation in a Low-resource Setting

Ken Yano and **Akira Utsumi**
The University of Electro-Communications
1-5-1 Chofugaoka, Chofu, Tokyo, JAPAN
yanoken@uec.ac.jp, utsumi@uec.ac.jp

Abstract

Because sign language is a visual language, the translation of it into spoken language is typically performed through an intermediate representation called gloss notation. In sign language, function words, such as particles and determiners, are not explicitly expressed, and there is little or no concept of morphological inflection in sign language. Therefore, gloss notation does not include such linguistic constructs. Because of these factors, we argue that sign language translation is effectively processed by taking advantage of the similarities and differences between sign language and its spoken counterpart. We thus propose a pipeline translation method that clearly focuses on the difference between spoken Japanese and signed Japanese written in gloss notation. Specifically, our method first uses statistical machine translation (SMT) to map glosses to corresponding spoken language words. We then use three transformer-based seq2seq models trained using a large out-of-domain monolingual Japanese corpus to complement postpositional particles and estimate conjugations for the verbs, adjectives, and auxiliary verbs in the first translation. We apply the seq2seq models in sequence until the translation converges. Our experimental results show that the proposed method performs robustly on the low-resource corpus and is +4.4/+4.9 points above the SMT baseline for BLEU-3/4.

1 Introduction

It is essential to build a social infrastructure for hearing-impaired and hearing communities to share sufficient information so that they can quickly obtain information necessary for daily life and disasters and lead a safe and secure life. Sign language used in deaf communities has different vocabulary and grammar from spoken language. There are two variations of sign language in Japan (Chonan, 2001): (1) Japanese Sign Language (JSL) and (2) Manually Coded Japanese (MCJ). JSL is often used by early signers, and syntax, such as word order and language structure, is different from spoken Japanese. By contrast, the syntax of MCJ is similar to that of spoken Japanese in terms of word order. It is used by late signers or acquired hearing-impaired people. However, the two variations are said to be used interchangeably and there is no clear boundary between them. In this work, we consider an intermediate between JSL and MCJ, and denote it Signed Japanese (SJ) in the following discussion.

Translation from sign language to spoken language is typically performed in two steps. First, consecutive signs are recognized from a video signal and transformed into an intermediate representation called a *gloss*, then the gloss is translated into a sentence in spoken language. Current state-of-the-art sign language recognition and translation methods (Camgöz et al. 2020; Yin and Read 2020) require a large amount of data and pay little attention to differences between sign language and the corresponding spoken language. Therefore, the success of these approaches relies heavily on large paired corpora, and resource-poor sign language studies, including SJ, cannot take advantage of such approaches. In sign language, function words, such as pre-positional or post-positional particles and determiners, do not tend to be explicitly signed, and inflectional morphemes associated with verbal predicates that express categories, such as tense, mood, and aspect, are not manually signed, in general. For example, in SJ, post-positional particles '助詞' are generally not explicitly signed, and the signs associated with verbal predicates are not conjugated, whereas in Japanese, verbs, adjectives, and auxiliary verbs are conjugated. Therefore, the gloss does not include such language constructs.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 2021–2032
August 1–6, 2021. ©2021 Association for Computational Linguistics

Although gloss notation is commonly used by writing a series of spoken words that correspond to each sign in capital letters, because of the lack of sign language resources, its quality and size differ greatly according to the language (Bungeroth et al., 2008). SJ signs are heavily polysemous and their meaning is often context sensitive. Additionally, there is no publicly available corpus for SJ translation studies. Therefore, in this study, we use an in-house corpus that uses our gloss notation method. The details of the corpus and its notation are described in Section 2.

To solve challenging problems, we propose a novel pipeline method to translate from SJ written in gloss to Japanese. In particular, we focus on the linguistic differences between SJ and Japanese and estimate the post-positional particles that are missing and the appropriate forms of morphological inflection of words. Our method assumes that the ground truth gloss of the signed sentence is available. This assumption does not limit the availability of the above two-step sign language translation method. Our method first uses phrase-based statistical machine translation to match the SJ gloss to Japanese words. Then we refine the results further using transformer-based seq2seq (Sutskever et al., 2014) models, which are trained using a large out-of-domain parallel corpus. Specifically, we use three different seq2seq models (1) to complement the post-positional particles, (2) to apply morphological inflection by conjugating verbs, adjectives, and auxiliary verbs, and (3) to re-estimate the post-positional particles over the previous output. We repeatedly apply these models sequentially and adjust the translation results until they converge.

The proposed method works robustly, even for small training datasets, which are typically of the order of thousands of pairs in a dataset, and the results show that the state-of-the-art method is inferior to the SMT baseline with the low-resource setting. We found that iterative updates of translations are effective for improving the grammaticality and fluency of the translation output. Our experimental results show that the proposed model provides +4.4/+4.9 higher translation performance for BLEU3/BLEU4 scores compared with the SMT baseline.

2 Materials

We use two corpora: one is a small in-house SJ and Japanese parallel corpus, and the other is a large out-of-domain Japanese monolingual corpus. We describe the details of each corpus as follows.

2.1 SJ and Japanese parallel corpus

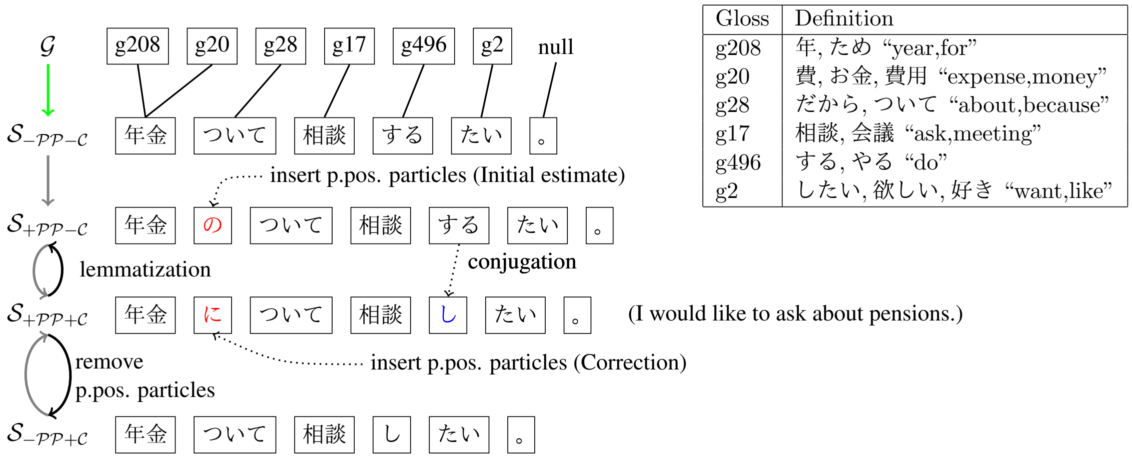
The locally organized in-house parallel corpus contains 1,086 sentence pairs with $>$7.5K glosses from a vocabulary of 655 words, and $>$11K Japanese words from a vocabulary of $>$1.2K words. The average length of a gloss sentence is 6.9 words, with a maximum length of 12 words and minimum length of 2 words, and the average length of a Japanese sentence is 10.3 words, with a maximum length of 21 words and minimum length of 5 words. The corpus consists of the ground truth gloss transcriptions of signs and their translations to Japanese sentences. The sentences are various spontaneous conversations that seemingly took place at municipal offices, such as asking for a certified copy of the resident register, pension, and unemployment insurance. In the corpus, a gloss word is written in the form gN, where N corresponds to an arbitrary unique number. We adopt this notation instead of using Japanese words because glosses in SJ are heavily polysemous and a sign maps to different Japanese words depending on the context. Instead, we use an auxiliary dictionary to map each gloss to spoken words or phrases. This notation method also helps the proposed method to select the appropriate Japanese word or phrase within the phrase-based statistical machine translation model that we use in the study.

Because of the sparsity of the parallel corpus, approximately 2.3% of the glosses are singletons, so we add all gloss dictionary items as additional parallel data to reduce OOV issues at test time.

2.2 Out-of-domain Japanese corpus

We use a subset of the Balanced Corpus of Contemporary Written Japanese [1] as an out-of-domain Japanese corpus to manually generate pseudo parallel corpora. The details of the corpus generation procedure are described in Section 3. To select the subsets, we use pattern matching to select sentences that end in a pattern such as [∼か？(question), ∼しました。(admit), ∼ほしいで

[1] https://pj.ninjal.ac.jp/corpus_center/bccwj/en/


$\mathcal{G}$
g208
g20
g28
g17
g496
g2
null
$\mathcal{S}_{-\mathcal{PP}-\mathcal{C}}$
年金
ついて
相談
する
たい
。
insert p.pos. particles (Initial estimate)
$\mathcal{S}_{+\mathcal{PP}-\mathcal{C}}$
年金
の
ついて
相談
する
たい
。
lemmatization
conjugation
$\mathcal{S}_{+\mathcal{PP}+\mathcal{C}}$
年金
に
ついて
相談
し
たい
。
(I would like to ask about pensions.)
remove p.pos. particles
insert p.pos. particles (Correction)
$\mathcal{S}_{-\mathcal{PP}+\mathcal{C}}$
年金
ついて
相談
し
たい
。

Gloss	Definition
g208	年, ため “year,for”
g20	費, お金, 費用 “expense,money”
g28	だから, ついて “about,because”
g17	相談, 会議 “ask,meeting”
g496	する, やる “do”
g2	したい, 欲しい, 好き “want,like”

Figure 1: Overall pipeline translation architecture from a gloss sequence $\mathcal{G}$ to a grammatical Japanese sequence $\mathcal{S}_{+PP+C}$. The green and gray arrows indicate the translations by a statistical machine translation model and three neural machine translation models, respectively. The table in the upper right corner shows an excerpt from the gloss dictionary. (p.pos. = post-positional)

す。(ask), ∼ います。(confirm), ∼ します。(intend), ∼希望する。(desire)]. These patterns were chosen so that the selected sentences are similar to the target Japanese in the paired corpus. The monolingual corpus contains >195K sentences with >3.9M Japanese words from a vocabulary of >70K Japanese words.

3 Methodology

The overall proposed pipeline translation system is shown in Fig. 1. $\mathcal{G}$ represents a gloss sequence and $\mathcal{S}$ represents a Japanese sequence. We define two subscripts for $\mathcal{S}$, that is, PP, which denotes ‘post-positional particle,’ and C, which denotes ‘conjugation,’ with the prefix $+$ or $-$ for each subscript, which denotes the existence or non-existence, respectively, of post-positional particles and conjugation. The definition of each term is provided in Table 1.

	post-positional particles	conjugation
$\mathcal{S}_{-PP-C}$		
$\mathcal{S}_{+PP-C}$	✓	
$\mathcal{S}_{-PP+C}$		✓
$\mathcal{S}_{+PP+C}$	✓	✓

Table 1: Definitions of terms for the variants of $\mathcal{S}$

3.1 Translation method

The proposed pipeline translation consists of six steps. Algorithm 1 shows the steps applied in sequence to gradually convert from $\mathcal{G}$ to $\mathcal{S}_{+PP+C}$, which is the final translation of this algorithm. The details of each step are as follows:

Step 0: Translate $\mathcal{G}$ into $\mathcal{S}_{-PP-C}$

We use a phrase-based statistical machine translation (pbsmt) (Koehn et al., 2007) to translate $\mathcal{G}$ into $\mathcal{S}_{-PP-C}$. In this step, we map each gloss phrase to the appropriate Japanese phrases without considering the post-positional particles and conjugations of the output Japanese sequence.

Step 1: Translate $\mathcal{S}_{-PP-C}$ into $\mathcal{S}_{+PP-C}$

We use a transformer-based seq2seq model (Vaswani et al., 2017) (s2s_m1) to translate $\mathcal{S}_{-PP-C}$ into $\mathcal{S}_{+PP-C}$. In this step, the model estimates missing post-positional particles in $\mathcal{S}_{-PP-C}$ and inserts them to generate $\mathcal{S}_{+PP-C}$.

Step 2: Translate $\mathcal{S}_{+PP-C}$ into $\mathcal{S}_{+PP+C}$

We use another transformer-based seq2seq model (s2s_m2) to translate $\mathcal{S}_{+PP-C}$ into $\mathcal{S}_{+PP+C}$. In this step, the model estimates the appropriate morphological inflection or conjugated form for verbs, adjectives, and auxiliary verbs in $\mathcal{S}_{+PP-C}$ to generate $\mathcal{S}_{+PP+C}$.

Step 3: Convert $\mathcal{S}_{+PP+C}$ to $\mathcal{S}_{-PP+C}$

In this step, we remove the previously estimated post-positional particles in $\mathcal{S}_{+PP+C}$ from Step 2 to generate $\mathcal{S}_{-PP+C}$.

Step 4: Translate $\mathcal{S}_{-PP+C}$ into $\mathcal{S}_{+PP+C}$

We use the other transformer-based seq2seq model (s2s_m3) to translate $\mathcal{S}_{-PP+C}$ into $\mathcal{S}_{+PP+C}$ by re-estimating missing post-positional particles. In Step 1, we estimated the post-positional particles over a Japanese sequence in the canonical form ($\mathcal{S}_{-PP-C}$); however, in the present step, we estimate them over for the conjugated word sequence ($\mathcal{S}_{-PP+C}$). We assume that this will correct the previous estimation using the conjugated form of the word sequence obtained in the previous steps.

Step 5: Convert $\mathcal{S}_{+PP+C}$ to $\mathcal{S}_{+PP-C}$

In this step, we transform $\mathcal{S}_{+PP+C}$ to $\mathcal{S}_{+PP-C}$ by converting words in $\mathcal{S}_{+PP+C}$ to their canonical form. Steps 3, 4, 5, and 2 are repeated until the translation output converges or the number of iterations reaches the maximum limit (10).

Algorithm 1 Translation of a sign gloss sequence into a Japanese sentence

```
Input: $\mathcal{G}$
Output: $\mathcal{S}_{+PP+C}$
  Step 0: $\mathcal{G} \rightarrow \mathcal{S}_{-PP-C}$
  Step 1: $\mathcal{S}_{-PP-C} \rightarrow \mathcal{S}_{+PP-C}$
  Step 2: $\mathcal{S}_{+PP-C} \rightarrow \mathcal{S}_{+PP+C}$
  $\mathcal{S}_{prev} = \emptyset$
  $\mathcal{S}_{next} = \mathcal{S}_{+PP+C}$
  while $\mathcal{S}_{prev} \neq \mathcal{S}_{next}$ do
      $\mathcal{S}_{prev} = \mathcal{S}_{next}$
      Step 3: $\mathcal{S}_{+PP+C} \rightarrow \mathcal{S}_{-PP+C}$
      Step 4: $\mathcal{S}_{-PP+C} \rightarrow \mathcal{S}_{+PP+C}$
      Step 5: $\mathcal{S}_{+PP+C} \rightarrow \mathcal{S}_{+PP-C}$
      Step 2: $\mathcal{S}_{+PP-C} \rightarrow \mathcal{S}_{+PP+C}$
      $\mathcal{S}_{next} = \mathcal{S}_{+PP+C}$
  end while
  return $\mathcal{S}_{next}$
```

4 Training

4.1 Statistical machine translation model in Step 0

We use Moses (Koehn et al., 2007) to train the phrase-based statistical machine translation model to translate from $\mathcal{G}$ to $\mathcal{S}_{-PP-C}$ in Step 0. To train the model, we use the parallel corpus and preprocess the target Japanese sequences by deleting post-positional particles, and convert conjugated words, such as verbs, adjectives, and auxiliary verbs, to their canonical forms using MeCab [2].

Note that we leave any post-positional particles untouched if gloss words corresponding to them exist. For example, the Japanese word か *ka* which is a post-positional particle bound to the end of an interrogative sentence, has a corresponding gloss word in SJ. The translation model is described by the following noisy-channel model to estimate the best target Japanese word sentence $s \in \mathcal{S}_{-PP-C}$ for a source gloss sentence $g \in \mathcal{G}$ as

$$s_{best} = argmax_s \; p(g|s)p_{LM}(s), \qquad (1)$$

where $p_{LM}(s)$ is a language model based on the n-grams of $\mathcal{S}_{-PP-C}$. $p(g|s)$ is decomposed into a phrase-based formula using a phrase translation table and phrase reordering model (Koehn et al., 2007). For the language model, we use modified 3-gram Kneser–Ney smoothing.

4.2 Encoder-decoder translation models in Steps 1, 2, and 4

For the seq2seq models used in Steps 1, 2, and 4 in Section 3.1, we use a transformer-based encoder-decoder model (Ott et al., 2019). For all models, we use an encoder and decoder with an embedding of size 512, FFN-embedding of size 2048, and six layers with eight attention heads. We use the Adam optimizer with label smoothing cross-entropy loss with a smoothing factor of 0.1. We set the initial learning rate to 5e-4 with a warmup updates of 4,000 and use the inverse sqrt learning rate scheduler. We set the maximum tokens in a batch to 4K. We use tied embedding for the input and output layers. We obtain the hyperparameters from a non-exhaustive parameter search, and the results are shown in Table 9 of the Appendix.

We randomly split the corpus into training, validation, and test sets in an 8:1:1 ratio. For tokenization, we use byte pair encoding (Sennrich et al., 2016), which is trained using the training dataset. We set the number of operations to 10K for each tokenization of the seq2seq models. For the seq2seq model in Step 1, we pre-process the out-of-domain monolingual corpus as follows:

- Source: preprocess Japanese sequences by deleting post-positional particles and converting all conjugated words, such as verbs, adjectives, and auxiliary verbs, to their canonical forms.
- Target: preprocess Japanese sequences by leaving post-positional particles untouched and converting all conjugated words, such as verbs, adjectives, and auxiliary verbs, to their canonical forms.

The pre-processed corpus becomes the pseudo-parallel corpus to train the seq2seq model in Step

[2] https://pypi.org/project/mecab-python3/

1 to translate $\mathcal{S}_{-PP-C}$ into $\mathcal{S}_{+PP-C}$. The training corpora of the other seq2seq models, that is, s2s_m2 in Step 2 and s2s_m3 in Step 4, are similarly preprocessed and independently trained using the pseudo-parallel corpus.

We observed that training the seq2seq model in Step 2 took more than a few hundred epochs, whereas training the seq2seq model in Steps 1 and 4 took less than 30 epochs. We used the models with the lowest validation loss for the experiments. The results on the test set demonstrated that the BLEU4 scores were 74.20, 98.75, and 75.06 for s2s_m1, s2s_m2, and s2s_m3, respectively. This indicates that post-positional particle estimation is more uncertain compared with the estimation of morphological inflection.

5 Experiments

To evaluate the proposed method, we conducted 100 experiments and for each test, we randomly selected 10 samples from the parallel corpus for testing and used the remaining samples to retrain the statistical machine translation model in Step 0. For each test, we finetuned the parameters of the seq2seq model (s2s_m1, s2s_m2, s2s_m3) by the training data and we used a beam size of 5 for decoding of the models. We averaged the 100 results to calculate the metrics of the performance.

We denote the proposed method in Section 3.1 by SMT+Iterative_s2s and compared its performance with the following baselines (naive, LSTMs, and SMT), the variants of the proposed method (SMT+1step_s2s and SMT+2step_s2s) and the transformer-based end-to-end Gloss2Text (G2T) model proposed by Yin and Read (2020). The followings are the brief explanations of each model.

- **naive**: This baseline replaces each gloss word with a Japanese word using the gloss dictionary. If more than one Japanese word is defined for a gloss, the first word is used.
- **LSTM**: This baseline uses encoder-decoder LSTM with an attention mechanism (Bahdanau et al., 2015) to directly translate $\mathcal{G}$ into $\mathcal{S}_{+PP+C}$. The model is trained using the parallel corpus without using the out-of-domain corpus and is configured with several different hyperparameter settings.
- **SMT**: This baseline uses only the statistical machine translation model to directly translate $\mathcal{G}$ into $\mathcal{S}_{+PP+C}$. This model is trained using the parallel corpus and without using the out-of-domain corpus.
- **SMT+1step_s2s**: This model is a variant of the proposed model which first executes Step 0 of Algorithm 1 to translate $\mathcal{G}$ into $\mathcal{S}_{-PP-C}$. Then it uses another seq2seq model trained using the out-of-domain corpus to directly translate $\mathcal{S}_{-PP-C}$ into $\mathcal{S}_{+PP+C}$. We compared the performance of this model, which jointly estimates post-positional particles and conjugations, with the model that estimates them separately using different models.
- **SMT+2step_s2s**: This model is another variant of the proposed model which performs Steps 0–2, but does not iteratively update the translation result as it does in Algorithm 1. We examined how the iterative updates of the result with **SMT+Iterative_s2s** contribute to the performance compared with the model without them.

For G2T, we changed the original hyperparameters suggested by Yin and Read (2020) and found that the following parameters were optimal using hyperparameter search on our parallel corpus. Encoder and decoder: embed-size = 256, FFN-embed-size = 1024, num-layer = 1, num-attention-head = 4.

5.1 Results

Table 2 shows the results of the experiment. To evaluate performance, we used the following metrics, BLEU-1/2/3/4 (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and word error rate (WER), and averaged the scores to obtain the results. The results showed that the proposed model (SMT+Iterative_s2s) outperformed the other models. The poor performance of the naive model indicates that the simple lookup method using the gloss dictionary did not produce successful results. LSTMs with different hyperparameters varying in the dimensions of the embedding and the hidden layers (256, 512, 1024) and the number of layers (1, 2) show the baseline performances to directly translate $\mathcal{G}$ into $\mathcal{S}_{+PP+C}$. Among them, the LSTM with the dimensions of the embedding and the hidden layers of 1024 and the number of layers of 1 showed the best performance. The best LSTM and the G2T were inferior

	embed	hidden	layers	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	WER
naive				0.184	0.049	0.007	0.002	0.139	0.801
LSTM	256	256	1	0.628	0.538	0.453	0.365	0.623	0.419
	512	512	1	0.692	0.608	0.527	0.442	0.688	0.349
	1024	1024	1	0.717	0.634	0.552	0.467	0.714	0.320
	256	256	2	0.448	0.314	0.222	0.154	0.408	0.616
	512	512	2	0.558	0.440	0.325	0.225	0.530	0.502
	1024	1024	2	0.590	0.466	0.332	0.209	0.556	0.466
G2T (Yin and Read, 2020)				0.695	0.640	0.592	0.535	0.708	0.305
SMT				0.788	0.724	0.663	0.599	0.800	0.233
SMT+1step_s2s				0.810	0.752	0.697	0.638	0.830	0.301
SMT+2step_s2s				0.811	0.756	0.701	0.642	0.829	0.245
SMT+Iterative_s2s				**0.817**	**0.762**	**0.707**	**0.648**	**0.833**	**0.216**

Table 2: Performance evaluations of naive, LSTMs, G2T (Yin and Read, 2020), SMT, SMT+1step_s2s, SMT+2step_s2s, and SMT+Iterative_s2s on BLEU1/2/3/4↑ and METEOR↑, and word error rate (WER) ↓ by averaging the scores from all experiments. The hyperparameters of LSTMs, the dimensions of the embedding and hidden layers, and the number of layers, are specified in the columns of 'embed', 'hidden', and 'layers', respectively. We use the same parameters for the encoder and the decoder of the LSTMs.

Model	Step	Source	Target	GS	EP
pbsmt	0	$\mathcal{G}$	$\mathcal{S}_{-PP-C}$	38.0	
s2s_m1	1	$\mathcal{S}_{-PP-C}$	$\mathcal{S}_{+PP-C}$	79.5	35.1
s2s_m2	2	$\mathcal{S}_{+PP-C}$	$\mathcal{S}_{+PP+C}$	99.2	35.1
s2s_m3	4	$\mathcal{S}_{-PP+C}$	$\mathcal{S}_{+PP+C}$	86.9	36.1

Table 3: Error propagation analysis of SMT+Iterative_s2s. The score is the exact match for the correct ratio (%) (GS = gold standard, EP = error propagation).

to the SMT because there were insufficient samples to train the neural models with large capacity. All the pipeline models that combined the SMT and seq2seq models outperformed the models that directly translate $\mathcal{G}$ into $\mathcal{S}_{+PP+C}$. This clearly demonstrates the effectiveness of the pipeline approach. Table 8 in Appendix illustrates the translation samples at each step of SMT+Iterative_s2s.

We investigated whether adding the monolingual Japanese corpus in 2.2 to train the target language model improved the performance of the SMT baseline. However, on the contrary, performance was slightly degraded. We believe that this was because of a domain mismatch between the corpora. The statistical significance test results confirmed that the performance of SMT+Iterative_s2s was significantly better than that of SMT, SMT+1step_s2s, and SMT+2step_s2s (see Table 10 in the Appendix).

Table 3 shows the error propagation analysis of SMT+Iterative_s2s. The score was measured using the exact match by counting the outputs that exactly matched the references at each step. Column 'GS' represents the gold standard score when using the ground truth input, and column 'EP' represents the score when using the output from the previous pipeline stage as the input propagating the errors. Clearly, a large portion of the error originated from Step 0 when translating $\mathcal{G}$ into $\mathcal{S}_{-PP-C}$. The GS score of s2s_m2 was much higher than that of s2s_m1 and s2s_m3, which was indicated by its higher BLEU score for the model evaluation on the test set described in Section 4. We verified that the EP score of s2s_m3 was 2.8% greater than that of s2s_m2, thereby illustrating the efficacy of the retrospective complement of post-positional particles. Note that the EP score of s2s_m3 was measured by allowing the output of s2s_m2 in the EP to be set to the input and removing all post-positional particles.

Table 4 shows the frequency of iterative update counts by SMT+Iterative_s2s. Approximately 72% of the results converged at the first iteration, and approximately 26% of the results converged at the second iteration. Counts above 6 were achieved when the same phrase was repeatedly generated, which is a phenomenon known as hallucination (Wang and Sennrich, 2020). If we detected such an error, we removed the repeating phrase to shorten the output.

5.2 Qualitative Evaluation

Table 5 shows the qualitative evaluations of the results using the proposed model and the other models with BLEU4, WER, and perplexity (PPL) scores. PPL in the last column was measured by the transformer-based language model that was pretrained by using the 494M-word Japanese Wikipedia.

We observed that the results of SMT and G2T had more post-positional particle selection errors than the other pipeline models, and the results of SMT+1step_s2s had more verb conjugation errors than SMT+2step_s2s, which suggest the efficacy of the independent estimation of post-positional particles and conjugations. We confirmed that the post-positional particle estimations using SMT+Iterative_s2s were either more natural or less error-prone than those using SMT+2step_s2s, which made the translation results more fluent.

Loop count	Freq.
1	723 (72.3%)
2	258 (25.8%)
3	4 (0.4%)
4	6 (0.6%)
6	1 (0.1%)
7	2 (0.2%)
8 ≥	6 (0.6%)

Table 4: Frequency of the iteration counts of Algorithm 1 until the translation output converged using SMT+Iterative_s2s.

Table 6 shows the average perplexity scores of the results of the SMT and pipeline models. While the perplexities of the pipeline models were much lower than that of SMT, the perplexity of the proposed SMT+Iterative_s2s was not the lowest. This result suggests that word-based perplexity is not suitable for evaluating equally acceptable translation outputs.

6 Discussion

In Table 5, most of the outputs using SMT+2step_s2s and SMT+Iterative_s2s were grammatically acceptable Japanese sentences with slight differences in the post-positional particle selections. As shown in the second and third examples in Table 5, the PPL scores of SMT+2step_s2s were lower than those of SMT+Iterative_s2s, but the BLEU4 and WER scores of SMT+Iterative_s2s were better than that of SMT+2step_s2s, even though the meanings of the sentences were almost the same. By contrast, the sentences of SMT+2step_s2s and SMT+Iterative_s2s in the first and last examples had different meanings, even though the PPL, BLEU4, and WER scores indicated that the results of SMT+Iterative_s2s were better than those of SMT+2step_s2s. However, depending on the context, the results of SMT+2step_s2s may be more appropriate. The main cause of the ambiguity issue is related to the information bottleneck raised by Yin and Read (2020) regarding the gloss notation of sign language. Currently, our parallel corpus does not include any non-manual signals (NMSs), such as facial expression, eye gaze, mouth, and movement of the head and shoulders. However, NMSs act as grammatical markings for syntactic information (Valli et al. 2011; Koizumi et al. 2002). NMSs are not expressed in sequence, but simultaneously with manual signs, and their subtleties make sign recognition and annotation more difficult. Perhaps, this is one of the reasons that most existing sign language corpora do not or only contain partial NMS labels along with glosses. As suggested by Yin and Read (2020), the performance of G2T translation may not impose an upper bound for sign-to-text translation unless the gloss faithfully describes the signed sentences. We are interested in investigating whether incorporating visual features from signs would improve the proposed G2T translation method. Because of the limited space in this paper, we leave this issue for future work.

Table 7 depicts examples of the translation errors by SMT+Iterative_s2s categorized into gloss word translation error, post-positional particle exchange, and conjugation exchange. As shown in Table 3, a large portion of the translation errors originated from gloss word translations. These errors mostly occurred because of the incorrect selection of Japanese wording for gloss phrases. For instance, the phrase 被災地 *hisaichi* ‘disaster area’ in the reference of the 2nd example in Table 7, which is expressed as a sequence of three glosses: 受ける ‘receive’, 災害 ‘disaster’, and 場所 ‘area’, was translated into the un-grammatical phrase, 受けて 災害場所 *ukete saigaibasho*. It is because the correct mapping from glosses to compound nouns cannot be learned by the phrase-based SMT unless they appear in the training set. The second major source of translation errors was the post-positional particle exchanges. These errors possibly change the semantic from the reference as indicated in the 5th example of Table 7, 図書館に本を寄贈する方法 “how to donate books to libray” v.s. 図書館の本を贈る方法 “how to give a book from the library”. As we mentioned above, some of these errors are difficult to handle because the system output may be correct in another context. Trans-

(1) Input $\mathcal{G}$	g467 g12 g77	BLEU4	WER	PPL
Reference	私の子供を探します。 “I look for my child.”			82.8
naive	私 子供 探す	0	0.714	117731.4
G2T	私 を 探し ます 。	0.474	0.286	119.7
SMT	私 子供 を 探し て い ます 。	0	0.429	59.3
SMT+1step_s2s	私 子供 探す ます 。	0	0.429	11715.6
STM+2step_s2s	私 も 子供 を 探し ます 。	0.643	0.143	129.6
SMT+Iterative_s2s	私 の 子供 を 探し ます 。	1	0	82.8
(2) Input $\mathcal{G}$	g19 g20 g87 g294 g20 g307 g202 g9	BLEU4	WER	PPL
Reference	医療 費の自己負担額は安くなりますか？ “Will my medical expense be cheaper?”			89.7
naive	医療 お金 自分 負担 お金 安い なる か	0	0.75	56747.3
G2T	医療 費 の 自己 負担 割合 は なり ます か ？	0.551	0.167	138.1
SMT	医療 費 の 自己 負担 の 安い に なり ます か ？	0.531	0.25	203.9
SMT+1step_s2s	医療 費 自己 負担 費 安い なる ます か ？	0	0.417	1223.0
STM+2step_s2s	医療 費 は 自己 負担 費 が 安く なり ます か ？	0.417	0.25	107.2
SMT+Iterative_s2s	医療 費 の 自己 負担 費 は 安く なり ます か ？	0.735	0.083	112.4
(3) Input $\mathcal{G}$	g73 g475 g52 g19 g27 g151 g2	BLEU4	WER	PPL
Reference	障害者向け医療支援を知りたい。 “I want to know medical support for disabilities.”			121.4
naive	障害 者 会う 医療 助成 わかる 欲しい	0	0.667	79281.5
G2T	障害 者 向け 医療 支援 を 知り たい 。	1	0	121.4
SMT	障害 者 向け 医療 支援 を 知り たい 。	1	0	121.4
SMT+1step_s2s	障害 者 向け 医療 支援 知る たい 。	0.525	0.222	768.0
STM+2step_s2s	障害 者 向け の 医療 支援 を 知り たい 。	0.658	0.111	62.7
SMT+Iterative_s2s	障害 者 向け 医療 支援 を 知り たい 。	1	0	121.4
(4) Input $\mathcal{G}$	g258 g860 g24 g8 g33 g9	BLEU4	WER	PPL
Reference	老人ホームの情報をいただけますか？ “Can you provide information about elderly housing with care ?”			155.8
naive	老人 住宅 情報 いただく できる か	0	0.667	4571.7
G2T	老人 ホーム の 情報 は いただけ ます か ？	0.597	0.111	148.3
SMT	老人 住宅 へ の 情報 を いただけ ます か ？	0.661	0.222	320.7
SMT+1step_s2s	老人 ホーム 情報 いただける ます か ？	0	0.333	943.4
STM+2step_s2s	老人 ホーム で 情報 を いただけ ます か ？	0.661	0.111	199.2
SMT+Iterative_s2s	老人 ホーム の 情報 を いただけ ます か ？	1	0	155.8

Table 5: Sample translation results by naive, G2T, SMT, SMT+1step_s2s, SMT+2step_s2s, and proposed SMT+Iterative_s2s. All the results of SMT+Iterative_s2s were when the iterative update converged in the second loop. PPL represents perplexity.

	Average perplexity
SMT	934.47
SMT+1step_s2s	241.44
SMT+2step_s2s	248.01
SMT+Iterative_s2s	269.92

Table 6: Average perplexity of translations using SMT, SMT+1step_s2s, SMT+1step_s2s, and SMT+Iterative_s2s measured by the transformer-based language model trained on the 494M-word Japanese Wikipedia.

lation errors relating to the conjugation exchange rarely occurred, and even if they did, the impacts were minimal.

7 Related works

Camgöz et al. (2018) proposed end-to-end sign language translation in the framework of neural machine translation, allowing them to jointly learn the spatial sign representation, underlying language model, and mapping between sign and spoken language using PHOENIX-Weather 2014T (Camgöz et al., 2018) corpus. Their later work (Camgöz et al., 2020) further improved the model by introducing a transformer-based architecture that jointly learns sign language recognition and translation while being trainable in an end-to-end manner using connectionist temporal classification loss to bind the recognition and translation problems into a single unified architecture.

In a similar research line, Yin and Read (2020) proposed the G2T model using a transformer-based seq2seq model, and evaluated the performance on PHOENIX-Weather 2014T (Camgöz et al., 2018) and ASLG-PC12 (Othman and Jemni, 2012) in various ways by changing the numbers of encoder-decoder layers and embedding schemes. All the end-to-end state-of-the-art sign language translation methods rely on large datasets and cannot be used for resource-poor datasets. The

Reference	System	Error type
どう すれ ば 留学 が でき ます か ？ “How can I study abroad?”	**どの よう に** 留学 でき ます か ？	gloss word translation error
被災 地 の ボランティア を し たい です 。 “I would like to volunteer in the disaster area.’	**受け て 災害 場所** を ボランティア に し たい です 。	gloss word translation error
災害 に 備え て 何 を 備蓄 し て い ます か ？ “What are you stockpiling in case of a disaster?”	災害 に 備え て どんな **保存** が あり ます か ？	gloss word translation error
タクシー 代 は 医療 費 控除 でき ます か ？ “Can taxi fare be deducted from medical expenses?”	タクシー 費 **や** 医療 費 控除 は でき ます か ？	post-positional particle exchange
図書館 に 本 を 寄贈 する 方法 を 教え て ください 。 “Please tell me how to donate books to the library.”	図書館 **の** 本 を 贈る 方法 を 教え て ください 。	post-positional particle exchange
名字 だけ の 印鑑 登録 でき ます か ？ “Is it possible to register a seal with only the surname?”	姓 だけ __ 印鑑 登録 は でき ます か ？	post-positional particle exchange
収入 が ない 人 で も 保険 料 を 払い ます か ？ “Do you pay insurance premiums even if you have no income?”	収入 が ない 人 **は** 保険 料 を 払う の です か ？	post-positional particle exchange
何 か いい 情報 は あり ます か ？ Do you have any good information?	何 が **よく** 情報 が あり ます か ？	conjugation exchange
休日 に 子供 を 預け られる ところ は あり ます か ？ “Is there a place where I can leave my child on holidays?”	休日 に 子供 を 預け **たり できる** ところ は あり ます か ？	conjugation exchange

Table 7: Examples of the translation errors by SMT+Iterative_s2s are categorized into gloss word translation error, post-positional particle exchange, and conjugation exchange. We highlight the **wrong** words or phrases in bold.

pipeline method that we proposed is related to the transfer learning method proposed by Mocialov et al. (2018). They proposed transfer learning to improve British Sign Language modeling at a gloss level by fine-tuning or layer substitution on neural network models pre-trained on the Penn Treebank dataset. Although the purpose of their work was not to translate sign language, their work is similar to ours in that it takes advantage of linguistic commonality between resource-poor sign language and its spoken language. Our approach to converting non-grammatical sentences into grammatical sentences is related to previous work on grammatical error correction (Imamura et al. 2012; Liu et al. 2018; Oyama et al. 2013). They used insert or replace operations to correct particle or morphological inflection errors in a monolithic model, and we believe that the proposed seq2seq-based iterative method using multiple models can be used for similar tasks.

8 Conclusion

We proposed a pipeline machine translation method from SJ to Japanese by assuming that the gloss of the sign is provided. We focused on grammatical differences between SJ and Japanese, particularly post-positional particles and morphological inflections, and proposed a pipeline model by cascading the phrase-based statistical machine translation and three transformer-based seq2seq models, which effectively addressed the resource-poor issue of the sign language corpus. The statistical machine translation model first maps each gloss phrase to a Japanese phrase, then three seq2seq models pre-trained using the monolingual corpus transform the initial translation by complementing post-positional particles, and apply conjugations for verbs, auxiliary verbs, and adjectives. Translation is repeated until the output converges. We confirmed that the proposed method outperformed the SMT baseline by +4.4/+4.9 points for BLEU-3/4.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback. This work was funded in part by the joint project “Research and development on a mutual communication system that connects hearing-impaired and hearing people” between the University of Electro-Communications and SoftBank Corp.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations ICLR*, San Diego, CA, USA.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Jan Bungeroth, Daniel Stein, Philippe Dreuw, Hermann Ney, Sara Morrissey, Andy Way, and Lynette van Zijl. 2008. The ATIS sign language corpus. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC’08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

N. C. Camgöz, S. Hadfield, O. Koller, H. Ney, and R. Bowden. 2018. Neural Sign Language Translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7784–7793.

N. C. Camgöz, O. Koller, S. Hadfield, and R. Bowden. 2020. Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10020–10030.

Hirohito Chonan. 2001. Grammatical Differences Between Japanese Sign Language, Pidgin Sign Japanese, and Manually Coded Japanese: Effects on Comprehension. *The Japanese Journal of Educational Psychology*, 49(4):417–426.

Kenji Imamura, Kuniko Saito, Kugatsu Sadamitsu, and Hitoshi Nishikawa. 2012. Grammar error correction using pseudo-error sentences and domain adaptation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 388–392, Jeju Island, Korea. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Atsuko Koizumi, Hirohiko Sagawa, and Masaru Takeuchi. 2002. An annotated Japanese Sign Language corpus. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

Jun Liu, Fei Cheng, Yiran Wang, Hiroyuki Shindo, and Yuji Matsumoto. 2018. Automatic error correction on Japanese functional expressions using character-based neural machine translation. In *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation*, Hong Kong. Association for Computational Linguistics.

Boris Mocialov, Helen Hastie, and Graham Turner. 2018. Transfer learning for British Sign Language modelling. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 101–110, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

A. Othman and M. Jemni. 2012. English-ASL Gloss Parallel Corpus 2012: ASLG-PC12.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Hiromi Oyama, Mamoru Komachi, and Yuji Matsumoto. 2013. Towards automatic error type classification of Japanese language learners' writings. In *Proceedings of the 27th Pacific Asia Conference on Language, Information, and Computation (PACLIC 27)*, pages 163–172, Taipei, Taiwan. Department of English, National Chengchi University.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. 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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, page 3104–3112. MIT Press.

Clayton Valli, Ceil Lucas, Kristin J. Mulrooney, and Miako N.P. Rankin. 2011. *Linguistics of American Sign Language*. Gallaudet University Press, Washington, DC.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc.

Chaojun Wang and Rico Sennrich. 2020. On exposure bias, hallucination and domain shift in neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3544–3552, Online. Association for Computational Linguistics.

Kayo Yin and Jesse Read. 2020. Better sign language translation with STMC-transformer. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5975–5989, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Appendix

A Sample translations by SMT+Iterative_s2s

Pipeline translation from $\mathcal{G}$ into $\mathcal{S}_{+PP+C}$	
$\mathcal{G}$	g6 g847 g71 g64 g84 g9
$\mathcal{S}_{-PP-C}$	図書館 どこ ある ます か ？
$\mathcal{S}_{+PP-C}$	図書館 は どこ に ある ます か ？
$\mathcal{S}_{+PP+C}$	図書館 は どこ に あり ます か ？
	“Where is the library ?”
$\mathcal{G}$	g25 g26 g27 g470
$\mathcal{S}_{-PP-C}$	補聴器 購入 助成 お願い する ます 。
$\mathcal{S}_{+PP-C}$	補聴器 の 購入 の 助成 を お願い する ます 。
$\mathcal{S}_{+PP+C}$	補聴器 の 購入 の 助成 を お願い し ます 。
	“Please subsidize the purchase of hearing aids.”
$\mathcal{G}$	g215 g555 g28 g181 g470
$\mathcal{S}_{-PP-C}$	入院 手続き ついて 教える くださる 。
$\mathcal{S}_{+PP-C}$	入院 の 手続き について 教える て くださる 。
$\mathcal{S}_{+PP+C}$	入院 の 手続き について 教え て ください 。
	“Please tell me about the procedure for hospitalization.”

Table 8: Examples of translation results at each pipeline step by SMT+Iterative_s2s. The words in red are the complemented post-positional particles and the words in blue are the conjugated words from the underlined lemmas above. (See the gloss definitions in Table 11.)

B Hyper-parameters tuning results

encoder & decoder				s2s_m1	s2s_m2	s2s_m3
embed-dim	ffn-embed-dim	layers	attention-heads	BLEU4	BLEU4	BLEU4
512	2048	6	8	**74.20**	**98.75**	**75.06**
256	1024	3	4	73.99	96.84	74.88
128	512	2	2	70.75	98.26	71.58
64	256	1	1	56.33	94.84	61.70

Table 9: The results of hyperparameter tuning of the seq2seq models (s2s_m1, s2s_m2, and s2s_m3) used in Steps 1, 2, and 4 of Algorithm 1, respectively. The values are BLEU4 scores of the test samples extracted from the parallel corpus.

C Statistical significance test results

	BLEU3			BLEU4		
	SMT	SMT+1step_s2s	SMT+2step_s2s	SMT	SMT+1step_s2s	SMT+2step_s2s
SMT+1step_s2s	1e-10**			4e-06**		
SMT+2step_s2s	1e-10**	0.164		1e-10**	0.177	
SMT+Iterative_s2s	1e-10**	0.003**	0.013*	1e-10**	0.018*	0.049*

Table 10: Non-parametric bootstrap test results of BLEU3 and BLEU4 for SMT, SMT+1step_s2s, SMT+2step_s2s, and SMT+Iterative_s2s. The values are the p-values. (‘*’ : p-value <0.05, ‘**’ : p-value <0.01)

D Snippets of gloss definitions

g12	子供, 児童 “child, children”	g467	私 “I”
g151	知る, わかる “know”	g470	お願い “ask, please”
g181	教わる “learn”	g475	者, 民 “man, people”
g19	医療 “medical”	g52	向け, 対象, 対する “to, about”
g2	したい, 欲しい, 好き “want, like”	g555	手続き “procedure”
g20	費, お金, 費用 “expense, money”	g6	本, 手帳 “book, memo”
g202	なる “become”	g64	場所 “place”
g24	情報 “information”	g71	何, どの, どれ “what, which, what”
g25	補聴器 “hearing aid”	g73	障害, 壊す “disability, damage”
g258	老人 “old, elderly”	g77	探す, 観光, 見学 “search, sightseeing”
g26	購入 “buy”	g8	いただく, もらう “receive, have”
g27	助成, 支援, 補助 “support, assistance”	g84	ある, です “be”
g28	だから, ので, ついて “because, about, in regard to”	g847	建物 “building”
g294	負担 “charge”	g860	住宅 “housing”
g307	安い “cheap”	g87	自分 “self”
g33	できる, 可能, よろしい “can, may”	g9	か？, あなた “yes/no, you”

Table 11: Definitions of glosses used in Tables 5 and 8.