Japanese-English News Translation in Different Scripts

Giang Le
Department of Linguistics
University of Illinois
gianghl2@illinois.edu

Abstract

This paper proposes a neural machine translation pipeline for the Japanese to English news task, using data augmentation techniques such as byte-pair encoding, adding voice information to the source sentences, and training with different Japanese scripts (standard orthography with mixed scripts, hiragana, and Romanized Japanese). The training, development, and test data come from Reuters and Paracrawl, provided by the WMT20 organizers. Source suffixes were appended to indicate the voice of the source language as it was anticipated that the use of voice would be important for news translation's fluency because it gives additional information about the discourse of the text. The preprocessed data were then trained with a Transformer Encoder-Decoder neural network model, considered to be the state-of-the-art in machine translation. The model was mplemented in the OpenNMT-py toolkit. The result was assessed against the test sets using the BLEU metrics.

1 Introduction

The language pair chosen for my machine translation news task is Japanese-English. This language pair is notably a challenging one because of the two languages' significant differences in word orders. While English is an inflectional SVO Germanic language, Japanese is an agglutinating SOV linguistic isolate. Japanese is also a high context language with frequent pro-drop strategies employed in conversations and an intricate system of politeness levels expressed mainly via verbal conjugations. For the news domain, I do not expect to encounter much of the pro-drop and politeness variation issues, but syntactic differences remain to be the major challenge in translating between Japanese and English. Additionally, encoding voice information of the source text could be crucial in improving news translation.

Two sets of data were used in this project for training, development, and testing purposes. A toy model was first trained on a small set of data, which comes from the Reuters corpus (Utiyama and Isahara, 2003), provided by Japan's National Institute of Standards and Technology (NIST). A part of this corpus was held out as a development set and a smaller set was used for testing. A full model was trained on a much larger set of data, the Paracrawl corpus. Sentence pairs with a score of equal or greater than 0.777 were extracted from the full Paracrawl corpus provided by the WMT20 organizers. A part of this corpus was also held out as a development set and a smaller set was used for testing.

In order to improve the quality of news translation for this language pair, the original plan was to incorporate a pre-ordering technique based on dependency parsing (Hoshino et al., 2013) and experimented training NMT systems with Japanese orthography in three forms, hiragana, mixed script, and Romanized Japanese. I planed to use the KNP dependency parser (Kurohashi, 1998) to generate dependency structures and POS tags for Japanese tokens and then apply pre-ordering steps to the data. This technique would comprise of four rules that reorder Japanese constituents to make the intermediate word order more alike to a head-initial structure, namely (1) pseudo head-initialization moves the verbal head element to the sentence's beginning, creating a VSO intermediate form, (2) inter-chunk pre-ordering converts the sentence to the SVO order, (3) inter-chunk normalization normalizes coordinate structures and punctuations, and (4) intra-chunk pre-ordering switches the order of particles and content words to create a head-initial structure within chunks. This preordering experiment, however, is not yet complete due to time constraints. Therefore, this report shall focus mainly on the result from training a Japanese to English translation task in different orthographies as well as the additional step where the voice information was added to the training data.

Other preprocessing steps involve tokenization, which was done using fugashi, a Python wrapper for the MeCab morphological analyzer. Another very important preprocessing step is learning byte pair encoding (Sennrich et al., 2016) as this step helps to reduce the vocabulary size. Also the parallel data were cleaned by removing sentences that were either too short or too long. The Moses scripts were used for English tokenization. I experimented with three settings for the Japanese texts, mixed scripts, hiragana only, and romaji only. I also added additional information about the voice to the source sentence during training, as translating voice and topics correctly can improve the fluency of news text. The sentence pairs were then trained with a Transformer Encoder-Decoder neural network model (Luong, 2015; Bahdanau et al., 2015), provided by the OpenNMT-py toolkit (Klein et al., 2017). Final evaluation of the translation result was assessed using the sacreBLEU toolkit and the BLEU metrics incorporated in OpenNMT-py.

2 Prior Work

The pre-ordering technique was among one of the techniques used to address the divergent syntax between English and Japanese, and it seemed to have had some success when used with phrase-based SMT. (Komachi et al., 2006) suggested a preordering approach for Japanese-English speech translation in the travel domain based on predicate-argument structure, which used a predicate argument structure analyzer and reordered Japanese sentences using heuristic rules. (Katz-Brown and Collins, 2008) presented patent translation results where the data were processed using two pre-ordering methods. Interestingly, their experiments showed that a naive reordering method where Japanese tokens were reversed performed better than a reordering method based on morphological analysis and dependency structure. Most recently, (Hoshino et al., 2013) proposed predicate-argument structure-based preordering rules in two levels for the Japanese-English patent translation task.

Previous NMT systems for Japanese-English have utilized different techniques to deal with various challenges that this language pair poses. To improve on the stylistics of the output, such as politeness (Senrich, 2016), Japanese honorifics (Feely, 2017), and voice (Yamagishi et al., 2016), target prefixes and source suffixes can be applied during training to add more meta-textual information (topic, domain, user personality, dialogue act) to the corpora. Prefix constraints can be used to alleviate issues of domain adaptation, sentence length control, unaligned target words, and bidirectional decoding (Shunsuke et al., 2017). I expect that voice would be important for the news genre, and propose to add voice information as source suffixes during training.

Another issue that NMT has to grapple with is the very high computational cost due to high dimensionality of the input and output layers, such that the size of the vocab often has to be restricted. There have been efforts to replace the UNK tokens denoting infrequent words by paraphrasing and replacing them with frequent words (Sekizawa et al., 2017), tracking the alignment of OOV tokens and replacing them via dictionary lookup or identity copy (Luong et al., 2015), selecting phrases that contain OOV tokens using branching entropy and then post-translating them using SMT phrase translation (Long et al., 2017), using a model that is not based on words but subwords or characters (Sennrich et al., 2016), or extracting OOV words by the attention mechanism of OpenNMT and then translating them with another NMT with a character-based model. I decided to apply the byte pair encoding technique to tackle the issue of vocab size, while also experimenting training with different Japanese scripts, such as *hiragana* only, or Romanized alphabet only.

Data availability is another issue for Japanese-English translation, albeit not as severe as some other language pairs. Efforts to augment the data include applying back translation (Sennrich et al., 2016) using monolingual data (Wang et al., 2017). Another technique that could help with data scarcity is domain adaptation via Mixed Fine Tuning (Chu et al., 2017; Dabre et al., 2019), which uses a combination of out-of-domain and in-domain corpora. Additional tags would often be used for a kind of domain-aware training. For the news domain, not a lot of in-domain training data are available in Japanese. The Reuter corpus is not particularly large, and the ParaCrawl corpus was mined from web sources and could contain out-of-domain data.

Other techniques that have been employed when working with NMT systems include implementing beam search and ensemble encoding (Matsumura and Komachi, 2017) and initializing the word embedding layers of the encoder and decoder with pre-trained embeddings that were previously trained on parallel corpora (Neishi et al., 2019).

3 Approach

This project consists of six experiments. Each experiment varies on how the source data in Japanese were preprocessed. Two factors varied, namely the choice of the Japanese orthographic script and whether voice information was added to the training data, giving rise to six ways of preprocessing the data: original mixed orthography, with voice; hiragana, with voice; romaji, with voice; mixed orthography, without voice; hiragana, without voice; and romaji, without voice.

Both Japanese and English datasets were normalized using Byte Pair Encoding (Sennrich et al., 2016) in order to reduce the vocabulary size, because high dimensionality is a concern for neural machine translation. Voice information was appended to the Japanese corpora by a matching heuristic because active and passive structures are frequent in news reports and are crucial in rendering fluent translations and fluid discourse.

subsectionData For the toy model, the Reuters parallel corpus was used for both training, development, and test set. This corpus consists of sentences from the English Reuters Corpus, Volume 1 and the Japanese sentences from the Multilingual Reuters Corpus, Volume 2. The file was encoded in EUC-JP originally and the sentences were aligned with a score indicating the quality of alignment (Utiyama and Isahara, 2003). EUC-JP stands for Extended Unix Code and is the Unix encoding for Japanese. It encodes most characters in two bytes. This corpus was converted to UTF-8 before any further preprocessing. The table below presents the statistics associated with the breakdown of the Reuters corpus into training, development, and test sets.

	Number of sentences
Train	50,000
Development	5000
Test	1,782

For the full model, the Paracrawl parallel corpus was used for both training, development, and test set. This corpus is the largest publicly available en-ja parallel corpus. The original corpus has about 10 million sentence pairs after deduplication. I extracted sentence pairs with an alignment score higher than 0.777 to include in the experiments. The table below presents the statistics associated with the breakdown of the remaining corpus into training, development, and test sets.

	Number of sentences
Train	1,292,000
Development	5000
Test	2,373

3.1 Method

3.1.1 Preprocessing for Japanese

The script conversion step was done using pykakasi, a tool for converting Japanese into different orthographic scripts. Japanese is written in a mixture of three different scripts: *kanji*, *hiragana*, and *katakana*. *kanji* means Chinese characters; it is used to write content words such as nouns, verb stems, adjectives, and so on. The Japanese Ministry of Education announced 2,136 *kanji* (jouyou kanji) that formed the kanji curriculum up to the high school level in Japan, as these characters are deemed to be the most regularly used. In more formal genres of writing, however, it is not uncommon to see *kanji* outside of this list. *hiragana* was derived from *kanji*. It is a phonetic syllabary, typically used to write conjugational endings, particles, and grammatical words. *katakana*, also a phonetic syllabary much like *hiragana*, is typically reserved to write foreign words, loan words, or strengthen the emotive content of the texts. In modern times, Japanese people are also familiar with the Latin alphabet due to exposure to English, and the Japanese language can be transliterated using an alphabet. This type of writing Japanese is called *romaji*. The sentence below shows how these three orthographic scripts are used together in one sentence:

東京都/kanji (proper name) は/hiragana (particle) 23/numeral 日/kanji (noun)、 新型/kanji (noun) コロナウイルス/katakana (loan word) の/hiragana (particle) 感染者/kanji (noun) を/hiragana (particle) 新/kanji (adjective) たに/hiragana (adverbial ending) 748/numeral 人/kanji (noun) 確認/kanji (verb) した/hiragana (verbal conjugation ending) と/hiragana (particle) 🄵 表/kanji (verb) した/hiragana (verbal conjugation ending).

It was confirmed that meta linguistic information such as politeness does not play a big role in the news genre, as the politeness markers hardly appear in the Reuters corpus. I tagged voice information to the data by looking up passive morphemes, which appear at the end of sentences and almost always written in hiragana. Three types of passive morphemes were identified and used as cue for the search:

- **Type 1** *rareru* appears after *godan* verbs, or verbs that end with *eru* or *iru*. This morpheme may appear in other forms to convey the past tense (*rareta*) or the progressive tense (*rareteiru*).
- **Type 2** *sareru* is the passive form of *suru*, a do-verb. It frequently occurs with Sino-Japanese nominal verbs.This morpheme may appear in other forms to convey the past tense (*sareta*) or the progressive tense (*sareteiru*).
- **Type 3** *areru* appears after *ichidan* verbs, or verbs that end with *i* in the stem (pre-masu) form. Because Japanese hiragana is a syllabic alphabet, all possible higarana's with *areru* endings were listed for matching.

Sentences identified as containing passive voice were then appended with a <Passive> tag. After this step, the data were tokenized using *fugashi* (McCann et al., 2010) based on an installed IPA dictionary. *fugashi* is a Python wrapper of MeCab, a C++ morphological analyzer for Japanese. MeCab analyzes Japanese morphology based on either the UniDic or the IPA dic. Ideally, the UniDic should be used as UniDic is more actively maintained, with new terms frequently being included.

3.1.2 Preprocessing for English

For the English data, data processing was conducted following the approach in the Sockeye paper (Hierber et al., 2018), namely:

- **Step 1** Normalize punctuation in the raw data using moses/scripts/tokenizer/normalize-punctuation.perl, specified for -l en
- **Step 2** Remove non printing characters using moses/scripts/tokenizer/remove-non-printing-char.perl, specified for -l en. No further character or casing normalization was performed.
- **Step 3** Tokenize by moses/scripts/tokenizer/tokenizer.perl, specified for the list of protected patterns included in Moses.

BPE (Sennrich et al. 2016) was learned from the tokenized data, by applying the subword-nmt package. Sentences were further normalized for lengths, where sentences longer than 100 tokens from the training data sets were removed, with the help of the clean-corpus-n.perl Moses script. A transformer encoder-decoder model was initialized through OpenNMT's implementation. I specified the configurations of the model with the adam optimizer, a learning rate of 2, and six layers are used for both encoder and decoder. The configurations are set to be the same between the encoder and the decoder.The configurations of model are updated in a .yaml file as shown below:

```
# Optimization
model_dtype: "fp32"
optim: "adam"
learning_rate: 2
warmup_steps: 8000
decay_method: "noam"
adam_beta2: 0.998
max_grad_norm: 0
label_smoothing: 0.1
param_init: 0
param_init_glorot: true
normalization: "tokens"

# Model
encoder_type: transformer
decoder_type: transformer
position_encoding: true
enc_layers: 6
dec_layers: 6
heads: 8
rnn_size: 512
word_vec_size: 512
transformer_ff: 2048
dropout_steps: [0]
dropout: [0.1]
attention_dropout: [0.1]
```

4 Results

Below are the results of the NMT Transformer model when trained with the Reuters corpus in the mixed script and hiragana, without any voice information. (It was intended that the toy Reuters data would be converted to romaji as well, but due to time constraints, this experiment has not been done). The training data in the mixed script and without voice information serve as the baseline for this report as no orthographic transformation nor voice information labelling were added to the data. Both results were quite dismal, due to the small size of the training data. The hiragana experiment gave a worse result compared to the baseline.

```
# Bleu score for mixed script Reuters,
no voice: 0.8
BLEU+case.mixed+numrefs.1+smooth.exp+
tok.13a+version.1.4.14 = 0.8
16.6/1.1/0.4/0.2
(BP = 0.722 ratio = 0.754
hyp_len = 34444 ref_len = 45669)

# Bleu score for hiragana Reuters,
no voice: 0.4
BLEU+case.mixed+numrefs.1+smooth.exp+
tok.13a+version.1.4.14 = 0.4
15.7/0.8/0.1/0.0
(BP = 0.805 ratio = 0.822
hyp_len = 37546 ref_len = 45669)
```

Results from the Reuters corpora, with voice information added, are presented below. Orthographic transformation was applied to the data, creating three experiments: mixed script, hiragana, and romaji. The sacrebleu scores show that training using romaji gave the best performance (average Bleu = 11.85), training using hiragana was slightly worse (average Bleu = 11.47) but using both scripts clearly gave better performances than training with the mixed script (average Bleu < 1.0).

BLEU	BP	Ratio
11.6	0.812	0.827
12.2	0.914	0.918
12.2	0.844	0.855
11.7	0.816	0.831
12.3	0.903	0.907
11.9	0.843	0.854
11.4	0.786	0.806
11.7	0.827	0.840
11.9	0.804	0.821
12.1	0.829	0.842
11.9	0.835	0.847
11.5	0.804	0.821
11.8	0.847	0.858
11.9	0.822	0.836
11.8	0.816	0.831
11.8	0.825	0.839
11.8	0.821	0.836
11.8	0.836	0.848
11.9	0.828	0.841
11.7	0.805	0.822

Table 1: Bleu scores over 20 training steps of romaji data

BLEU	BP	Ratio
11.6	0.800	0.818
10.3	0.795	0.814
11.2	0.787	0.807
12.1	0.854	0.864
11.8	0.836	0.848
11.7	0.808	0.825
11.9	0.822	0.836
11.2	0.792	0.811
11.3	0.767	0.791
11.3	0.786	0.806
11.3	0.799	0.817
11.6	0.807	0.823
11.9	0.795	0.813
11.4	0.770	0.793
11.5	0.773	0.795
11.5	0.785	0.805
11.4	0.759	0.784
11.4	0.777	0.799
11.5	0.803	0.820

Table 2: Bleu scores over 20 training steps of hiragana data

BLEU	BP	Ratio
0.7	0.789	0.809
0.7	0.758	0.783
0.7	0.783	0.803
0.8	0.769	0.792
0.7	0.765	0.788
0.6	0.804	0.821
0.7	0.781	0.802
0.8	0.829	0.842
0.7	0.780	0.801
0.7	0.776	0.798
0.8	0.782	0.803
0.8	0.791	0.810
0.7	0.786	0.806
0.8	0.780	0.801
0.8	0.791	0.810
0.7	0.766	0.789
0.8	0.766	0.789
0.8	0.772	0.795
0.7	0.773	0.795

Table 3: Bleu scores over 20 training steps of mixed script data

Below are results of the NMT Transformer model when trained with the Paracrawl corpora in different scripts and with the voice information included. Similarly to the Reuters experiment, when the training data were transformed to the hiragana and romaji script, the score improved compared to when the training data remained in the original mixed script. The Bleu scores ranking in descending order are hiragana script, romaji script, and mixed script.

```
# Bleu score for mixed script Para: 19.9
BLEU+case.mixed+numrefs.1+smooth.exp+
tok.13a+version.1.4.14 = 19.9
43.6/23.1/15.0/10.5
(BP = 1.000 ratio = 1.127
hyp_len = 62824 ref_len = 55731)

# Bleu score for hiragana Para: 21.5
BLEU+case.mixed+numrefs.1+smooth.exp+
tok.13a+version.1.4.14 = 21.5
44.5/24.5/16.4/12.0
(BP = 1.000 ratio = 1.125
hyp_len = 62713 ref_len = 55731)

# Bleu score for romaji Para: 20.8
BLEU+case.mixed+numrefs.1+smooth.exp+
tok.13a+version.1.4.14 = 20.8
44.4/23.9/15.7/11.3
(BP = 1.000 ratio = 1.139
hyp_len = 63500 ref_len = 55731)
```

5 Discussion

The improved Bleu score of the hiragana experiment when voice information was included (11.47) compared to the hiragana experiment presented when no voice information was included (0.4) suggests that including voice information in the training data clearly improved the performance of this news translation task, as this comparison was made on the same dataset (the Reuters corpus). Unfortunately, due to time constraints, I was not able to obtain results of the baseline experiment in different scripts using the bigger Paracrawl dataset. The fact that the increase in the Bleu score is very high when the voice information tag was included in the training data is encouraging to this round of experiments. This is also in line with the result reported in other studies that have attempted to augment the training data with contextual information such as politeness levels. Formality-aware NMT models were reported to perform better than baseline models in a previous study (Feely, 2017) and other experiments in English-German translation (Senrich et al, 2006) also showed that ”the choice of honorifics has a big impact on translation quality as measured by BLEU ... substantial improvements are possible by constraining the translation to the desired level of politeness”. More similarly to what was attempted in this study are the experiments conducted in (Yamagishi, 2016), which yielded an improvement of 0.73 Bleu points when using gold voice labels.

The larger data set (Paracrawl) in general shows better performance than the smaller Reuters corpus, which is expected. The choice of script affects the Bleu scores, with the hiragana and romaji scripts outperforming the mixed script most of the times, except for the Reuters corpus without voice labels (The hiragana experiment gives a worse result compared to the mixed script, by 0.4 points). It is unclear why this is the case. Better performances when the training data were presented in different orthographies are notable and could be explained by the different vocabulary sizes that were obtained when the vocabularies were built. Smaller vocabulary sizes are more advantageous because they reduce computational cost of a model. Demand for smaller vocabulary spurred the development of algorithms and techniques to reduce the vocabulary sizes. One of the most popular approaches is subword training mentioned earlier. In the case of Japanese, training using the hiragana and romaji script might have contributed positively to a smaller vocabulary size. The vocabulary size obtained from the mixed script experiment is 34626, much larger than the vocabulary size obtained from the hiragana (27843) and romaji (27428).

A technique that has been used in phrase-based statistical NMT for the Japanese-English pair is reordering of the Japanese text. As Japanese and English have very different underlying syntactical structures, bringing them closer together would be a step towards reducing the complexity of prediction in the model. To implement preordering, the sentences should parsed by some type of syntactical parser, such as a dependency parser. Constituents such as the final Verbal phrase should be moved up to emulate the English SVO order. For example, the token labeled as ROOT would first be moved to the beginning of the sentence to simulate a SVO order and then intra-reordering would be needed to convert the sentences into head-initial chunks. It would be interesting to investigate how preordering would impact performance, along side other techniques that have been tried here.

6 Conclusion

This report experimented with training a NMT Transformer encoder-decoder model to translate news data from Japanese to English using two main data augmentation techniques. Transforming the Paracrawl training data from a mixed script orthography to hiragana or romaji resulted in a drastic improvement in Bleu scores, likely due to a smaller vocabulary size. It was also found that adding contextual information to the training data such as including the voice feature of the sentences (Passive) resulted in an improvement. Further research is needed to complete the baseline training for the Paracrawl data, as well as examining ways to re-order the source sentences to be closer to the target's syntactic structures.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. Cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation.

Alexandre Berard, Ioan Calapodescu, Marc Dymetman, Claude Roux, Jean-Luc Meunier, and Vassilina Nikoulina. 2019. Machine translation of restaurant reviews: New corpus for domain adaptation and robustness. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 168–176, Hong Kong. Association for Computational Linguistics.

Jingsheng Cai, Masao Utiyama, Eiichiro Sumita, and Yujie Zhang. 2014. Dependency-based pre-ordering for Chinese-English machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 155–160, Baltimore, Maryland. Association for Computational Linguistics.

Raj Dabre and Eiichiro Sumita. 2019. NICT's supervised neural machine translation systems for the WMT19 translation robustness task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 533–536, Florence, Italy. Association for Computational Linguistics.

Weston Feely, Eva Hasler, and Adrià de Gispert. 2019. Controlling Japanese honorifics in English-to-Japanese neural machine translation. In *Proceedings of the 6th Workshop on Asian Translation*, pages 45–53, Hong Kong, China. Association for Computational Linguistics.

Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2018. Sockeye: A toolkit for neural machine translation.

Sho Hoshino, Yusuke Miyao, Katsuhito Sudoh, and Masaaki Nagata. 2013. Two-stage pre-ordering for Japanese-to-English statistical machine translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1062–1066, Nagoya, Japan. Asian Federation of Natural Language Processing.

Jason Katz-Brown and M. Collins. 2008. Syntactic reordering in preprocessing for japanese → english translation: Mit system description for ntcir-7 patent translation task. In *NTCIR*.

Mamoru Komachi, Y. Matsumoto, and M. Nagata. 2006. Phrase reordering for statistical machine translation based on predicate-argument structure. In *IWSLT*.

Julia Kreutzer, Jasmijn Bastings, and Stefan Riezler. 2019. Joey NMT: A minimalist NMT toolkit for novices. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 109–114, Hong Kong, China. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Yukio Matsumura and Mamoru Komachi. 2017. Tokyo metropolitan university neural machine translation system for WAT 2017. In *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, pages 160–166, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Paul McCann. 2020. fugashi, a tool for tokenizing japanese in python.

Masato Neishi, Jin Sakuma, Satoshi Tohda, Shonosuke Ishiwatari, Naoki Yoshinaga, and Masashi Toyoda. 2017. A bag of useful tricks for practical neural machine translation: Embedding layer initialization and large batch size. In *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, pages 99–109, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Graham Neubig. 2011. The Kyoto free translation task. http://www.phontron.com/kftt.

Graham Neubig, Yosuke Nakata, and Shinsuke Mori. 2011. Pointwise prediction for robust, adaptable Japanese morphological analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 529–533, Portland, Oregon, USA. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Controlling politeness in neural machine translation via side constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 35–40, San Diego, California. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016c. 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.

Masao Utiyama and Hitoshi Isahara. 2003. Reliable measures for aligning Japanese-English news articles and sentences. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 72–79, Sapporo, Japan. Association for Computational Linguistics.

Hayahide Yamagishi, Shin Kanouchi, Takayuki Sato, and Mamoru Komachi. 2016. Controlling the voice of a sentence in Japanese-to-English neural machine translation. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 203–210, Osaka, Japan. The COLING 2016 Organizing Committee.