Domain-independent Punctuation and Segmentation Insertion

Eunah Cho[*], *Jan Niehues, Alex Waibel*

Institute for Anthropomatics and Robotics,
Karlsruhe Institute of Technology,
Karlsruhe, Germany
firstname.lastname@kit.edu

Abstract

Punctuation and segmentation is crucial in spoken language translation, as it has a strong impact to translation performance. However, the impact of rare or unknown words in the performance of punctuation and segmentation insertion has not been thoroughly studied. In this work, we simulate various degrees of domain-match in testing scenario and investigate their impact to the punctuation insertion task.

We explore three rare word generalizing schemes using part-of-speech (POS) tokens. Experiments show that generalizing rare and unknown words greatly improves the punctuation insertion performance, reaching up to 8.8 points of improvement in F-score when applied to the out-of-domain test scenario. We show that this improvement in punctuation quality has a positive impact on a following machine translation (MT) performance, improving it by 2 BLEU points.

1. Introduction

Punctuation and segmentation for automatic speech recognition (ASR) output is crucial in order to provide a better readability of the transcript as well as for a better performance in a subsequent application, such as machine translation (MT). Current state-of-the-art ASR systems often do not generate any or reliable punctuation marks. Thus, there has been an extensive amount of study on this issue.

A widely used method for punctuation and segmentation insertion utilizes language model with prosody features [1] due to its low latency. On the other hand, translation model-inspired systems [2, 3, 4] show an outstanding performance, both in accuracy of punctuation marks and improving the following MT performance. Using such monolingual translation systems, a non-punctuated source language is *translated* into a punctuated source language. Recently a neural machine translation (NMT)-based model [5] is shown to have a better performance, maintaining low latency in a real-time application.

While the monolingual translation system for punctuation insertion has been thoroughly investigated for its performance in subsequent applications, such as MT, and for real-time scenario constraints, such as latency and compactness of the model, domain-dependency of the model and its potential impact have been left under-explored.

In this paper, we investigate the domain-dependency of punctuation and segmentation insertion task and suggest that a generalization scheme over domain-specific words can greatly improve the performance. In this scheme, rare and unknown words are represented in their part-of-speech (POS) tokens for generalization.

In order to analyze the problem, we consider three scenarios where a different amount of matching in-domain data is available for training. In the first scenario, test data and training data are from the same resource. Therefore they share a same genre. In the second scenario, we consider a case where only a small amount of in-domain data is available for training the punctuation insertion model. In the last scenario, no matching in-domain data is available for training. Detailed data description for each scenario will be given in Section 5.

We then design three different schemes for modeling rare-words in punctuation insertion, whose details of the schemes will be given in Section 4.

For the punctuation systems, we use an attentional encoder-decoder model [6], so that a non-punctuated text is punctuated and true-cased using a translation framework. The punctuation insertion system is built for two source languages, English and German. We also translate punctuated test data into another language and measured the translation performance, in order to evaluate the impact of punctuation insertion in a further down text processing.

Our experiments show that generalizing rare and unknown words for punctuation and segmentation insertion task brings up to 8.8 points of improvements in F-score. Experiments on both manual and ASR transcripts show that generalizing rare and unknown words using POS tokens improves punctuation accuracy and also enhances the performance of following MT.

This paper is organized as follows. In Section 2, we overview past research in related fields. The problem statement and motivation as well as a detailed description of the task are given in Section 3. In Section 4 we will describe how

[*] Now in Amazon: eunahch@amazon.com

rare words can be generalized for better performances and the scenarios we consider in this work. Section 5 describes three different domain-match scenarios in testing scenario. Section 6 contains a detailed description of experimental setups, data preparation and evaluation settings, followed by Section 7 where the results and analyses are given. The paper is concluded in Section 8.

2. Related Work

Many previous research has been devoted to insertion of punctuation and segmentation into ASR transcripts. In [1], authors investigated using language model for the task, incorporated with prosody information such as pause duration. Authors in [7] explored maximum entropy model for the task using lexical as well as prosodic features. Modeling punctuation marks was also viewed as a sequential tagging problem in [8].

Punctuation insertion task is considered as a part of translation task in [2], where the authors build an implicit translation model, which translates non-punctuated source language into punctuated one. Later this work is extended by the authors in [3]. In this work, authors compared three approaches to view the punctuation insertion task as a machine translation problem. Among them, the explicit model where punctuation marks are inserted on the source side, prior to the translation, showed the best performance. This approach, however, can only be used under the assumption that the sentence boundary is already defined. Therefore, this approach would punctuation marks within pre-defined sentence boundaries only.

In [4], authors solved this problem by preparing the training data differently. They altered the training data so that sentence breaks are inserted in random locations. Therefore, sentence breaks can be observed anywhere throughout the data, not necessarily after a sentence-finalizing punctuation marks. On the other hand, they used a sliding window for testing.

Punctuation insertion task using neural networks has been studied using various architectures. The authors in [9] used a classifier based on a recurrent neural network (RNN). It is shown that a bidirectional recurrent network with attention mechanism can be effective for the task as well [10].

Recent development of attention-based NMT [6] has improved the performance machine translation greatly. An attentional NMT system consists of an encoder representing a source sentence and an attention-aware decoder that produces the translated sentence. In [5], a neural machine translation model is used as a method to insert punctuation marks into a non-punctuated source language. Authors investigated into the trade-off between network size and performance. By applying compact representation on the target side, they show that the NMT-based model outperforms PBMT-based model, maintaining low latency in an end-to-end translation scenario.

Domain adaptation and topic-matching problem for machine translation has been studied from various perspectives. In [11], authors gave a thorough analysis on different approaches to adapt a statistical machine translation system towards a target domain, using a small in-domain data. Techniques for domain adaptation in NMT has been explored and evaluated in an evaluation campaign in [12]. Rare word problem of NMT and potential solutions for machine translation scenario have been discussed in various literatures [13, 14]. Also, in [15] authors investigated how to adapt existing NMT systems to into a spoken language domain.

3. Domain-dependency of Punctuation and Segmentation Insertion Task

Domain adaptation for machine translation has received a great deal of attention [16], since applying an MT system into a test data of a different domain significantly affects translation quality.

In this paper, we study the impact of domain mismatch in the punctuation insertion task. Table 1 shows three separate excerpts extracted from a test data, which is punctuated using an NMT-based punctuation and segmentation system [5]. The system is trained on generic data and the test data contains domain-specific terminologies.

Table 1: *Three excerpts from test data, punctuated using a segmenter trained on generic data. Company and product names are anonymized.*

1	...use your existing Git and Gerrit Implementations. As well.
2	...server level should ever reference. The schema itself this.
3	...that might be an existing *#Company #Product1*. *#Product1-cont* system an, *#Product2* System it. Could be a replicated...

We can observe that the system provides rather poor quality of punctuation especially around rare words. Especially in the third excerpt, the product name (marked as *#Product1*), which originally consists of two tokens, is even separated by the inserted full stop.

Building separate domain-matching systems and obtaining a substantial amount of training data for each domain is costly. Therefore, we aim to build a punctuation insertion system which can be used relatively independent from the domain of test data. In order to generalize rare words, we explore methods using POS tags. The details are described in Section 4.

4. Modeling of Rare Words

In this work, punctuation and segmentation insertion task is considered as translation problem. Lower-cased text without any punctuation is translated into true-cased text with proper punctuation and segmentation. While the NMT-based punctuation and segmentation insertion system shows a good result [5], the performance can be affected by rare words, as discussed in Section 3.

4.1. Definition

In this work, we define *rare word* as a word occurring less than 10 times throughout the training corpus. Additionally, we also need to model unknown words during training, in order to account them during the test case. Thus, we define *unknown word* as a word occurred only once in the training data.

4.2. Model

For generalization of rare words, we utilize POS information in order to consider syntactic information of them. In this work, we compare three different methods to represent rare and unknown words in a generalized form using POS.

- *unknown-NN*: Only unknown words are generalized. In order to generalize unknown words, we replace all words that occurred only once in the training data, into a POS-tag for noun (NN). We choose NN as it is the most frequently occurring POS throughout the corpus.

- *rare-NN*: Rare words, including unknown words, are generalized into the POS-tag for noun (NN).

- *rare-MF*: Unknown words are mapped into a POS-tag for noun (NN), while rare words are mapped into each word's most frequently (MF) used POS tag. Thus, we build a MF map from the training corpus. The MF map stores the most frequently used POS for each unique word in the training data. We obtain the POS-tag for each word of the training corpus using TreeTagger [17].

Test data is prepared in a similar manner for each criteria. Unknown word, that was not observed during training, is replaced into *NN*.

An excerpt from the training data is shown in Table 2 to depict *rare-MF* operation. In the first example, we can see that a rare word *thrives* is replaced into its most frequent tag, *VVZ* for the source side. In the same way, words like *beryllium*, *Adit*, or *tungsten* in the second example are replaced into POS tags.

Once rare words are generalized using different methods, further preprocessings are applied in order to form a parallel data for MT training. Details are discussed in Section 6.2.

5. Scenarios

While an extensive amount of previous research investigate the punctuation insertion task [3, 5], the impact of non-matching domain in test case is under-explored. In order to establish the importance of domain-match for this task, we model three scenarios of in-domain data availability by utilizing test data and training data from different sources. We utilize in-house English and German data for different scenarios.

5.1. Matching Data

The first scenario simulates the case where we have enough genre-matching training data. We take the training data and test data from the same source and model and evaluate the punctuation prediction system on the English TED data[1].

The training data comprises of ∼200K sentences of TED corpus, while the in-domain test data is around 1K sentences of TED. The audio reaches around 2 hours and 16 minutes.

By modeling this scenario, we aim to evaluate the impact of generalizing rare words into POS tokens in the punctuation prediction system, even when it is applied to a perfectly genre-fitting input.

5.2. Small In-domain Data

In the next scenario, we consider the case where only a limited amount of in-domain data is available. The model is trained using around ∼200K sentences of German TED data concatenated with 10K sentences of lecture corpus [18]. While the lecture corpus may share a similar style with the TED corpus (monologue, lecture), the lecture corpus contains a variety of domain-specific terms. The punctuation insertion system is then tested on a lecture data. Its manual transcript has 3K sentences, and its audio reaches around 6 hours and 32 minutes.

Detailed analysis on the data statistics is shown in Table 3. In the top two rows, we show the word count information in the original corpus, before we replace rare words into POS tokens. In the third line, we show how many words in the training/test data (among all occurrences) are considered as rare words according to the definition given in Section 4.1. We can see that around 4.5% of words of training data are rare words. About 2.0% of words in training data has occurred only once throughout the corpus. In the lecture test data, around 3.1% of words are unknown words, which were not observed during the training. As the university lecture corpus contains domain-specific terminologies, we can see that overall 4.9% of words in the test data are rare words. When using *rare-MF* method to generalize the rare and unknown words, the training data has 14.7K unique words. The number for test data is also decreased to 3.4K.

5.3. No in-domain data

In this scenario, we evaluate the English punctuation insertion built on the TED data, described in Section 5.1, on an online lecture corpus obtained from an internal project. The manual transcript reaches around 700 sentences, with the audio of a length of 1 hour and 55 minutes. The english online

[1] `https://www.ted.com`

Table 2: *POS replacement for rare and unknown word generalization*

Original	... a type of bacteria that thrives at 180 degrees. I think that's ...
rare-MF	... a type of bacteria that VVZ at 180 degrees. I think that's ...
Original	it doesn't have any beryllium in it. it's called the Pole Adit. and it does have tungsten, ...
rare-MF	it doesn't have any NN in it. it's called the Pole NP. and it does have NN, ...

Table 3: *Data statistics: German*

	Train	Test
All word	3,866.2K	53.6K
Unique word	137.2K	6.2K
Rare word	4.51%	4.87%
Unknown word	1.95%	3.07%

lecture mostly covers its most recent technologies. Consequently the test data contains a relatively higher proportion of rare and unknown words. By applying the system on this out-of-domain test data, we aim to show the effectiveness of our system handling rare words.

Table 4: *Data statistics: English*

	Train	Test:in	Test:out
All word	3,801.5K	20.3K	17.8K
Uniq word	63.5K	3.1K	1.8K
Rare word	2.63%	2.73%	4.54%
Unknown word	0.68%	1.07%	4.67%

Data statistics for English training and two test data are summarized in Table 4. First two rows, same as before, are showing general statistics of training and test data before the POS-replacement operation was applied. We can see that the ratio of rare words to all words in the corpus for both training and in-domain test data is around 2.7%. However, this ratio for out-of-domain test data rises to 4.5%. More importantly, the out-of-domain data has a significantly higher ratio of unknown word, 4.7%, compared to training as well as the in-domain test data. The statistics shows that the out-of-domain test data indeed includes a great proportion of unknown and rare-words, which is replaced into POS tokens during the replacement operation. Using *rare-MF* system, the training data has 11.9K unique words, in-domain test data 2.5K and out-of-domain test data 1.3K unique words respectively.

6. Experimental Setups

Since this process already decreases the vocabulary size effectively, we did not use any sub-word units. The detailed data statistics changed by this process will be discussed in Section 5.

In this section, we discuss the architecture of NMT-based punctuation insertion system as well as machine translation systems used to translate the punctuated test data.

6.1. Punctuation Insertion by NMT-based System

Inspired by [5], all punctuation insertion systems are built using the NMT framework `lamtram` [19], with an attention-based encoder-decoder model.

The models were all trained with Adam, where the algorithm is restarted twice and early stopping is applied. In [5], authors investigated the tradeoff between network size and performance. Following this work, we also configured that the encoder uses word embeddings of size 128 and a bidirectional LSTM [20] with 64 hidden layers for each direction. We use a multi-layer perceptron with 128 hidden units for the attention. For the decoder, we use conditional GRU units with 128 hidden units. Both networks are applied with dropout at every layer with the probability of 0.5.

6.2. Data Preparation

General data preparation follows the work in [4]. Except for tokenization and true-casing, no other preprocessing is applied for both input languages. The training data is randomly cut so that sentence boundaries can be observed in any location throughout the segment. Source side of the training data consists of lower-cased words and/or POS tokens. All punctuation marks are removed.

In this work, we build three systems to measure the impact of generalizing rare words. The details of the generalization scheme is given in Section 4.2. Since generalization of rare words largely decreases the number of unique words in the training data, we did not apply any sub-word operations on the training data for the three systems.

As a *baseline* system, we build a system following the work in [5], where no POS-replacement operation is applied. Source words are instead applied with byte-pair encoding of an operation size 40K. As another comparative system, in addition, we build a system *all-MF* where all words are replaced into their most frequent tag. In this system, thus, source side text consists of POS tags only. For English *all-MF* system, we introduce an additional POS tag for the word *I*. Since this word is always uppercased in English, we believe that it is fair to introduce a separate token for it. Vocabulary size for the *all-MF* system is therefore same as the number of possible POSs in each language.

For all systems with different source side representation, the target side follows the compact representation shown in [5]. Therefore, the target side corpus consists of *U* (meaning to be uppercased token), *L* (to be lowercased), and punctuation marks. As punctuation marks, we only consider sentence boundary marks (.?!) and commas.

Test data is prepared in the same manner of training data, where random line breaks are inserted, in order to simulate ASR output.

6.3. Evaluation

As discussed in Section 5, English system, trained only on TED corpus, is evaluated on two different test sets, in-domain and out-of-domain test data. German system, whose training data includes a small lecture corpus, is tested on a lecture data.

The performance is measured intrinsically as well as extrinsically. The accuracy of punctuation marks inserted into manual transcripts is measured in F-score. Later this set is translated into another language in order to measure the impact of punctuation marks in translation performance. German lecture data is translated into English, and English TED data is translated into German. English lecture data is translated into Spanish following reference translation availability. The detailed description of each machine translation system used is given in Section 6.4.

The general system description for ASR is given in [21]. The training data for the English ASR system includes 450 hours of TED data. In addition, it also includes 30 hours of lecture courses obtained from the same project.

6.4. Machine Translation Systems

Punctuated German test data is translated into English for performance evaluation. The detailed description to German to English machine translation system can be found in [22, 23]. It is a phrase-based machine translation system that is trained on European Parliament and News Commentary corpora and adapted into TED and lecture domain.

The in-domain English test data, once it is punctuated, is translated into German. We use a phrase-based MT system described in [24]. The out-of-domain English test data has a Spanish reference translation. In order to evaluate the impact of punctuation prediction in this data, we use a neural machine translation system.

The English to Spanish NMT system is trained using the toolkit OpenNMT [25][2], with its default architecture. The training data includes EPPS, NC, and TED corpora, where a BPE of operation size 40K is applied. Additionally, we also use a data from Wikipedia[3] in order to support translation of domain-specific words [26].

The impact of inserted punctuation marks on each test data are measured in translation performance, in case-sensitive BLEU [27].

7. Results and Analysis

In this section, we show the results of experiments, followed by detailed analysis.

7.1. German Punctuation Insertion

In order to simulate scenarios with small in-domain training data, we build a system on German in-house data. Table 5 shows the results for German manual transcript.

Table 5: *Punctuation insertion performance: German lecture manual transcript*

System	F-score	De→En (BLEU)
Baseline	50.18	22.01
(1) unknown-NN	55.55	22.22
(2) rare-NN	55.23	22.30
(3) rare-MF	**56.79**	**22.61**
all-MF	47.21	21.25

When using *rare-MF* system to insert punctuation and segmentation into manual transcript, we can see that we achieve 6.6 points of F-score improvement over the *baseline*. This improvement also led to better translation, yielding 0.6 points of BLEU improvement by simply using different punctuation and segmentation into the same manual transcript prior to translation. As comparison, we also show the number of *all-MF*, where we can see the negative impact of over-generalization of this system.

Table 6: *Punctuation insertion performance: German lecture ASR transcript*

System	De→En (BLEU)
Baseline	18.71
(1) unknown-NN	19.12
(2) rare-NN	19.16
(3) rare-MF	**19.23**
all-MF	18.53

Table 6 also shows how much we can improve the translation of ASR transcript when we use the punctuation insertion system which generalizes rare words. Compared to the *baseline* where no generalization is applied, we improve the translation performance by 0.5 BLEU points. Thus, we can observe that the rare words in the lecture test data can be handled better when we use the *rare-MF* system.

When comparing the performance on manual transcripts and ASR, we see that in both cases rare-MF leads to the best performance. Also, in both cases, we improve the translation performance by 0.5 BLEU points.

7.2. English Punctuation Insertion

Table 7 shows the performance for English manual transcripts, tested on both in-domain test data and out-of-domain

[2] https://github.com/OpenNMT/OpenNMT-py
[3] https://wikipedia.org

test data. As mentioned, we show the intrinsic performance of punctuation insertion in F-score for each test data and extrinsic performance in BLEU. In-domain test data is translated into German, while out-of-domain test data, lecture test set, is translated into Spanish.

Table 7: *Punctuation insertion performance: English in-domain and out-of-domain manual transcript*

System	in-domainTest		out-domainTest	
	F-score	En→De	F-score	En→Es
Baseline	53.95	18.46	51.21	22.73
(1) unknown-NN	57.40	18.77	59.87	24.45
(2) rare-NN	59.23	**19.16**	57.93	24.45
(3) rare-MF	**59.63**	18.93	**59.99**	**24.68**
all-MF	38.10	16.87	42.82	20.89

We can observe that the *rare-MF* system yields a big improvement in F-score, 5.7 for in-domain test data and 8.8 for out-of-domain test data. This improvement in the F-score is also continued in the translation performance. The improvement in translation performance is bigger for out-of-domain test data, reaching around 1 BLEU point. The results show that generalizing rare and unknown words does not only improve the punctuation insertion but also the sequential applications' performance. The over-generalization of the additional system *all-MF* shows a worse performance for both test sets.

Table 8: *Punctuation insertion performance: English in-domain and out-of-domain ASR transcripts (BLEU)*

System	in-domainTest	out-domainTest
	En→De	En→Es
Baseline	13.74	19.21
(1) unknown-NN	**13.92**	20.22
(2) rare-NN	13.74	20.13
(3) rare-MF	13.77	**20.23**
all-MF	12.44	17.85

We apply the same set of experiments for English ASR transcripts. The results are shown in Table 8. For ASR transcripts, we translate the punctuated output into different languages and measure the performance in BLEU. As shown in the table, we can see that punctuation inserted from the *rare-MF* system into the in-domain test data did not yield a big performance improvement over the Baseline. For the out-of-domain test data, however, the *rare-MF* improves the translation performance around 0.8 BLEU points, by inserting the different punctuation marks and sentence segmentation only. The results show the importance of rare word generalization in the punctuation insertion system.

Another constant observation is that the more we lack of in-domain training data, the bigger improvement we may expect from using the *rare-MF* system.

7.3. Analysis

In addition, we measure the impact of using POS tokens over rare words in overall speed.

Table 9: *Running Time*

	English	German
Baseline	0m56.219s	1m49.818s
rare-MF	0m47.242s	1m45.493s
all-MF	0m42.341s	1m36.889s

The results are shown in Table 9. We measure the time used for decoding English in-domain test data and German test data, both manual transcripts. It is worth to note that the test sets are decoded in a CPU. We can see that decreasing vocabulary on the source side by replacing rare words into their POS tags, while keeping the target side vocabulary the same, overall testing time is decreased by 85$\sim$90% of the baseline system. Faster runtime is therefore another advantage of generalization of rare words, which is often crucial in real-time applications.

Table 10 shows excerpts from the test data of the scenario where we have only a little amount of in-domain training data. We can observe that while the baseline system often misplaces a punctuation mark, *rare-MF* offers a better performance.

8. Conclusion

In this work, we showed that the performance of punctuation and segmentation can be greatly improved by generalizing rare and unknown words. In order to evaluate the impact of this system, we set three different scenarios on in-domain data availability. Our experiments show that we can improve the F-score by 5.7 points even for the scenario where we have a perfectly genre-matching training data. In the setting where in-domain data is not available at all and therefore rare/unknown words occur very frequently, F-score was improved by 8.8 points and subsequently 1 BLEU point in the following translation task of the punctuated test data.

In a detailed data analysis, we show that using this generalization also decreases source vocabulary dramatically. Compared to the baseline where we use sub-word units, the vocabulary size is decreased to 30$\sim$37%. This also boosts faster running time during the testing.

Future work includes combining this model with other post-processing tasks for ASR, i.e. disfluency removal.

9. Acknowledgements

The project leading to this application has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement n° 645452. This work was supported by the Carl-Zeiss-Stiftung.

Table 10: *Excerpts from output using different segmentation and punctuation system*

Excerpt 1	Baseline	wir sind nur zehn Kilometer voneinander. entfernt mit einem Auto fünfzehn Minuten.
	En. gloss	we are only ten kilometres from each other. away with a car fifteen minutes.
	rare-MF	wir sind nur zehn Kilometer voneinander entfernt mit einem Auto, fünfzehn Minuten.
	En. gloss	we are only ten kilometres from each other away with a car, fifteen minutes.
Excerpt 2	Baseline	Universitäten sind bottom-up. strukturiert Ideen entstehen in kleinen Ecken ...
	En. gloss	Universtities are bottom-up. structured ideas grow in small corners...
	rare-MF	Universitäten sind Bottom-up strukturiert. Ideen entstehen in kleinen Ecken...
	En. gloss	Universities are bottem-up structured. Ideas grow in small corners ...

10. References

[1] S. Rao, I. Lane, and T. Schultz, "Optimizing Sentence Segmentation for Spoken Language Translation," in *Proceedings of the eighth Annual Conference of the International Speech Communication Association (INTERSPEECH 2007)*, Antwerp, Belgium, 2007.

[2] M. Paulik, S. Rao, I. Lane, S. Vogel, and T. Schultz, "Sentence Segmentation and Punctuation Recovery for Spoken Language Translation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2008)*, Las Vegas, Nevada, USA, April 2008.

[3] S. Peitz, M. Freitag, A. Mauser, and H. Ney, "Modeling Punctuation Prediction as Machine Translation," in *Proceedings of the eight International Workshop on Spoken Language Translation (IWSLT 2011)*, San Francisco, California, USA, 2011.

[4] E. Cho, J. Niehues, and A. Waibel, "Segmentation and punctuation prediction in speech language translation using a monolingual translation system," in *Proceedings of the International Workshop for Spoken Language Translation (IWSLT 2012)*, Hong Kong, China, 2012, pp. 252–259.

[5] ——, "Nmt-based segmentation and punctuation insertion for real-time spoken language translation," in *Interspeech*, 2017.

[6] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2015.

[7] J. Huang and G. Zweig, "Maximum Entropy Model for Punctuation Annotation from Speech," in *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)*, Denver, Colorado, USA, 2002.

[8] W. Lu and H. T. Ng, "Better punctuation prediction with dynamic conditional random fields," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2010)*. Cambridge, Massachusetts, USA: Association for Computational Linguistics, 2010, pp. 177–186.

[9] M. Kazi, B. Thompson, E. Salesky, T. Anderson, G. Erdmann, E. Hansen, B. Ore, K. Young, J. Gwinnup, M. Hutt, and C. May, "The MITLL-AFRL IWSLT 2015 systems," in *Proceedings of the Eleventh International Workshop on Spoken Language Translation (IWSLT 2015)*, Da Nang, Vietnam, 2015.

[10] O. Tilk and T. Alumäe, "Bidirectional recurrent neural network with attention mechanism for punctuation restoration," *Interspeech 2016*, pp. 3047–3051, 2016.

[11] J. Niehues and A. Waibel, "Detailed analysis of different strategies for phrase table adaptation in smt," in *Proceedings of the Tenth Conference of the Association for Machine Translation in the Americas (AMTA)*, 2012.

[12] E. Cho, J. Niehues, T.-L. Ha, M. Sperber, M. Mediani, and A. Waibel, "Adaptation and combination of nmt systems: The kit translation systems for iwslt 2016," in *IWSLT*, Seattle, WA, USA, 2016.

[13] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany, 2015.

[14] M.-T. Luong, I. Sutskever, Q. V. Le, O. Vinyals, and W. Zaremba, "Addressing the rare word problem in neural machine translation," *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2015.

[15] M.-T. Luong and C. D. Manning, “Stanford neural machine translation systems for spoken language domains,” in *Proceedings of the International Workshop on Spoken Language Translation*, 2015.

[16] P. Koehn and J. Schroeder, “Experiments in domain adaptation for statistical machine translation,” in *Proceedings of the second workshop on statistical machine translation*. Association for Computational Linguistics, 2007, pp. 224–227.

[17] H. Schmid and F. Laws, “Estimation of Conditional Probabilities with Decision Trees and an Application to Fine-Grained POS Tagging,” in *Proceedings of the 22nd International Conference on Computational Linguistics, Proceedings of the Conference (COLING 2008)*, Manchester, UK, 2008.

[18] S. Stüker, F. Kraft, C. Mohr, T. Herrmann, E. Cho, and A. Waibel, “The kit lecture corpus for speech translation,” in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, Istanbul, Turkey, 2012, pp. 3409–3414.

[19] G. Neubig, “lamtram: A toolkit for language and translation modeling using neural networks,” http://www.github.com/neubig/lamtram, 2015.

[20] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] T.-S. Nguyen, M. Mueller, M. Sperber, T. Zenkel, K. Kilgour, S. Stueker, and A. Waibel, “The 2016 kit iwslt speech-to-text systems for english and german,” in *IWSLT*, Seattle, WA, USA, 2016.

[22] E. Cho, C. Fügen, T. Herrmann, K. Kilgour, M. Mediani, C. Mohr, J. Niehues, K. Rottmann, C. Saam, S. Stüker, and A. Waibel, “A real-world system for simultaneous translation of german lectures,” in *Proceedings of the 14th Annual Conference of the International Speech Communication Association (INTERSPEECH 2013)*, Lyon, France, 2013.

[23] M. Müller, T. S. Nguyen, J. Niehues, E. Cho, B. Krüger, T.-L. Ha, K. Kilgour, M. Sperber, M. Mediani, S. Stüker, *et al.*, “Lecture translator speech translation framework for simultaneous lecture translation,” *NAACL HLT 2016*, p. 82, 2016.

[24] I. Slawik, M. Mediani, J. Niehues, Y. Zhang, E. Cho, T. Herrmann, T.-L. Ha, and A. Waibel, “The KIT Translation Systems for IWSLT 2014,” in *Proceedings of the eleventh International Workshop for Spoken Language Translation (IWSLT 2014)*, Lake Tahoe, California, USA, 2014.

[25] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush, “OpenNMT: Open-Source Toolkit for Neural Machine Translation,” *ArXiv e-prints*, 2017.

[26] J. Niehues and A. Waibel, “Using wikipedia to translate domain-specific terms in smt,” in *Proceedings of the eighth International Workshop on Spoken Language Translation (IWSLT 2011)*, San Francisco, California, USA, 2011.

[27] 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 on Association for Computational Linguistics (ACL 2002)*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, 2002, pp. 311–318.