Continuous Learning in Neural Machine Translation using Bilingual Dictionaries

Jan Niehues
Department of Data Science and Knowledge Engineering (DKE)
Maastricht University, Maastricht, The Netherlands
jan.niehues@maastrichtuniversity.nl

Abstract

While recent advances in deep learning led to significant improvements in machine translation, neural machine translation is often still not able to continuously adapt to the environment. For humans, as well as for machine translation, bilingual dictionaries are a promising knowledge source to continuously integrate new knowledge. However, their exploitation poses several challenges: The system needs to be able to perform one-shot learning as well as model the morphology of source and target language.

In this work, we proposed an evaluation framework to assess the ability of neural machine translation to continuously learn new phrases. We integrate one-shot learning methods for neural machine translation with different word representations and show that it is important to address both in order to successfully make use of bilingual dictionaries. By addressing both challenges we are able to improve the ability to translate new, rare words and phrases from 30% to up to 70%. The correct lemma is even generated by more than 90%.

1 Introduction

Recent advances in neural machine translation (NMT) have led to astonishing translation quality of research systems in evaluation campaigns as well as for commercial systems. These improvements even led to discussions whether automatic machine translation is already on par with human translation (Barrault et al., 2019). One challenge that has raised less attention is the ability of these systems to continuously learn over time. In contrast, humans are continuously improving their skills and adapting to an ever-changing environment.

There are several reasons why this is necessary: First, nobody is fluent in all possible domains. Even professional translators need to adapt to the specific vocabulary of different domains. Secondly, language is not static but developing over time and translators need to learn new terms, meanings and expressions.

For humans, one successful approach to adapt to the environment is the usage of a dictionary [1]. Learning translations from a dictionary has several advantages: Dictionaries contain minimal examples. We do not need to collect full sentences, but can directly learn translations from a single phrase. Furthermore, this can even be generalized to other inflected forms of the same lexem. Secondly, it enables the system to directly integrate correction. If a user sees a specific problem, the user can interact with the system by adding a specific dictionary entry. This is very important if a specific terminology should be used.

Motivated by the success for human translators, in this work we will enable NMT to also successfully integrate knowledge from bilingual dictionaries. Thereby, we will focus on learning translations that could not be learned from parallel data. This poses several interesting research challenges as shown in the example in Table 1. When training a system on the proceedings of the European Parliament, it might never have seen the word *giraffe* and needs to learn the translation from the dictionary. First of all, we have to address one-shot learning. The system needs to be able to continuously learn new dictionary entries and then should directly be able to translate all occurrences of this phrase.

Secondly, the model must be aware of the morphology of the source and target language. In a dictionary only the base form of a word is given. In the example only the lemma *giraffe* is in the dictionary, but not the plural form *giraffes*. Therefore, we must enable the system to translate different lexemes of a lemma by knowing only the translation of the base form. This involves analysing the mor-

[1] In this work the dictionary entries can consist of a single word or whole phrases

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 830–840
April 19 - 23, 2021.

Source:	Tell us, what have you got against **giraffes** ?
Dictionary:	giraffe → Giraffe
Reference:	was haben Sie eigentlich gegen **Giraffen** ?
Annotation:	Tell us, what have you got against # giraffes # Giraffe # ?

Table 1: Example of dictionary usage

phological form of the source word, transferring the information about the form to the target and finally generating the correct morphological form of the target word based on the dictionary entry as well as on the morphological form of the source word. In German the plural of the dictionary entry *Giraffe* is *Giraffen*.

In order to assess the approaches on this challenging condition, it is essential to define an appropriate evaluation scheme. While the ability to continuously learn new translations is essential in many practical applications, the newly learned terminology will only occur rarely. Therefore, standard methods for evaluating machine translation are not able to measure the effect appropriately.

In order to address these challenges, we develop the following contributions:

- We developed a targeted evaluation approach for the continuous learning of new translations (Section 2)
- We showed that character-based representation is essential to inflect unknown words correctly. (Section 3)
- We show that only the combination of word representation and one-shot learning enables the successful integration of bilingual dictionaries (Section 3)

2 Evaluation scenario

The first important research question that needs to be addressed in the targeted continuous learning scenario is the evaluation approach. While the evaluation of machine translation is well-established (e.g. using BLEU (Papineni et al., 2002)), new learned words are typically rare words and therefore their influence on a BLEU score calculated on all words is very limited.

In order to have a valid evaluation approach, the evaluation should focus on phrases that cannot be learned from the parallel data. These are typically very rare phrases. Furthermore, we want to translate them in a real world situation. Therefore, the evaluation data should not be synthetic sentences. Finally, the approach should be using the standard parallel data without the need of collecting additional parallel data.

A first attempt would be to use existing test data and select sentences where dictionary entries are needed as e.g. done in (Dinu et al., 2019). However, if we limit ourselves to phrases that do not occur in the parallel data or only a few times, the number of occurring words in the test sets are too low to draw any conclusions.

Therefore, we evaluate our approach by proposing a new test-train split of existing parallel data. In a first step, we filter a large background dictionary for entries that help to translate phrases that only occur a few times in the existing parallel data. In a second step, we select some of the sentences with their matching dictionaries entries as the new test sets. An overview of the process is shown in Figure 1. Finally, we specifically evaluate the ability of the translation system to translate the dictionary entries.

In addition, it is important to ensure that the proposed methods do not have negative side effects on the overall translation quality. Therefore, we also evaluate the model using standard evaluation metrics on well-established test sets and on the proposed test set. Due to the weakness of these metrics to measure improvements in rare words, we do not expect that the proposed methods improve on these metrics, but it is important that the performance measured in these metrics does not decrease significantly.

2.1 Dictionary filtering

In a first step, we create a large background dictionary for each considered language pair by extracting a bilingual dictionary from the English Wiktionary. Therefore, we extracted the translation from a Wiktionary dump[2] using wiktextract [3].

Secondly, we match the dictionary entries to the

[2]https://dumps.wikimedia.org/enwiktionary/20200501/enwiktionary-20200501-pages-articles.xml.bz2

[3]https://github.com/tatuylonen/wiktextract

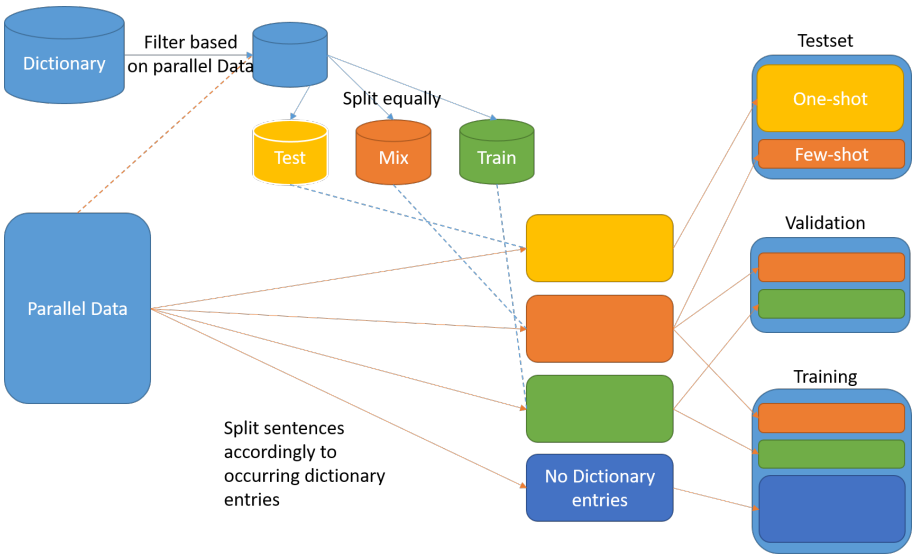

Figure 1: Overview of the evaluation approach: Based on the parallel data and dictionary, a new split of the data is generated

targeted corpus. Therefore, we lemmatize all dictionary entries as well as both sides of the parallel data. This is done to also find matches for all morphological variants of the dictionary entries. Finally, we calculate the statistics mentioned in Table 2 for each dictionary entry about its matches to the parallel data.

In a third step, we filter the dictionary based on the statistics. We only select words that are rare in the corpus. If the words are common and occur often in the training data, a dictionary entry would not be helpful. Secondly, we want to analyse the ability of the system to generate different morphological forms. Therefore, we only consider entries that occur at least with two different morphological variants on the target side. Finally, in this work we focus on words that are not ambiguous. We leave an integration of word sense disambiguation to also handle ambiguous dictionary entries for future work. Therefore, we only consider phrases, where both, the source and target phrase, occur less than 10 times with a different translation than the one given in the dictionary.

Statistic	Threshold
Occurrences	$3 \leq k \leq 80$
target inflected phrases	≥ 2
only source/target match	< 10

Table 2: Dictionary filtering

2.2 Train-Test Split

Finally we generate a split of the corpus into training, validation and test sets based on the selected dictionary entries as shown in Figure 1. The model needs to learn how to use the dictionary. Therefore, several training sentences need to be annotated with dictionary entries. Furthermore, we want to evaluate the ability of the model to translate phrases it has seen a few times in training (*Few-Shot learning*) as well as words it only has seen in the dictionary (*One-shot learning*). Therefore, we split the entries in the dictionary equally into three sets (*Test (yellow)*, *Mix (orange)* and *Train (green)*). All sentence pairs associated with entries from the *Test* set are added to the newly created test set.

In a second step, we select all the sentences from the remaining training sentences, where an entry from the *Mix* set occurs. For each entry, half the sentences are added to the test set and a quarter to the validation and training set.

Finally, all sentences with entries from *Train* are equally distributed to the training and validation set. Since we want to concentrate on modelling the morphology when using the dictionary and not the translation ambiguity for dictionary entries, we removed all sentences from the training where the source entry from the dictionary occurs, but the target sentence does not contain the target entry. Due to our selection of the dictionary, where we focus on words that have only very few different translations (less than 10 times a different one), we only removed very few sentences here. All

remaining sentences with no annotations (most of the sentences) were used for training.

2.3 Evaluation

When evaluating we want to focus on the system's ability to translate the phrases from the dictionary. Therefore, we measure the accuracy of translating the dictionary entries in addition to the commonly used BLEU score. In addition to calculating the accuracy by comparing the inflected words of hypothesis and reference (*Exact match*), we calculate further statistics to analyse the approaches.

In addition, we measure the ability of the system to at least create the correct lemma by ignoring errors made due to wrong inflection of the words. Therefore, for each sentence, we compare the target lemmatized phrase of the dictionary entry with the lemmatized version of the generated translation. We will refer to this metric as *Lemma match*

Finally, we are especially interested in the ability of the model to generate the correct inflected form. For many words, this is quite straightforward since it is the same as the lemma. Therefore, we also measure the exact match on the subset of the dictionary entries, where the target side of the dictionary is different from the inflect form occurring in the reference. To generate the correct translation, in this case the model really needs to change the output. We will refer to this metric as *Morph. Adjustment*.

In addition to these three evaluation scores, we also investigate the performance on the different types of entries. We evaluate all metrics on all entries and independently on the one-shot (*OneS*) and few-shot (*FewS*) entries.

3 NMT Dictionary Integration

To successfully integrate the dictionary into the NMT system, we need to address two challenges: First, we need to enable the system to perform one-shot learning. It should be able to translate a phrase after seeing it only once in the bilingual dictionary. Furthermore, it needs to be possible to continuously add new translations. Secondly, we need to model the morphology of the dictionary entries. We need to use the dictionary for different inflected forms of the word and also generate various inflected forms of the target phrase.

3.1 One-shot learning

In order to achieve one-shot learning, we need to combine the dictionary with our neural machine translation system. The combination should ensure fast learning, so a single dictionary entry is enough to learn the translation. Furthermore, it needs to be flexible, so new dictionary entries can be continuously added to the system and it is able to perform life-long learning by using the newly added entries.

One large advantage of deep learning approaches is that they are able to easily incorporate additional information. By annotating the input with additional information, the model is able to learn automatically how to make use of this additional information. This has been successfully done, for example, for the translation of other MT systems (Niehues et al., 2016), for domain information (Kobus et al., 2017) or information about formality (Sennrich et al., 2016a).

For the integration of additional knowledge about specific phrases, we follow similar approaches presented in Pham et al. (2018) and Dinu et al. (2019). The main idea is that we annotate each source phrase, for which a dictionary translation is available with this translation. This is done by appending the translation to the source phrase within the sentence as shown in Table 1. Since this is done during training and testing, the system is able to learn to copy and modify these suggestions. No further adaptation to the architecture of the NMT system is necessary. The system will learn how to exploit these systems and can transfer this knowledge to new translations that have not been seen in training. Therefore, the translations need only to be added once to the dictionary, which enables the system to perform one-shot learning as well as to continuously learn new translation by extending the dictionary.

The main difference to previous work is that we are focusing on very rare words and morphological variants of the dictionary phrases. Therefore, we investigate the matching of the dictionary entries as well as the number of necessary entries.

In order to find the dictionary entries for a given source sentence, we first lemmatize the sentence. In a second step, we then match the dictionary to the lemmatized sentences. Finally, we map back the found entries to the original sentence.

In annotating the source sentence, we follow the related work and append the translation to the source phrase. As shown in Figure 1, we replace

the source word *giraffes* by the entry *# giraffes # Giraffe #*. In contrast to the original work, we do not have the inflect target words, but only put the lemmatized target string to the sentence. For the source side, we keep the inflected form for the source sentence so the system is able to extract important morphological information from the source (e.g. grammatical number) and map it to the target. This is done for the training and test data. Then the baseline neural machine translation system is trained normally on the annotated sentences. We did not adapt the architecture since in Dinu et al. (2019) the standard transformer based system was able to learn to copy the suggested translations into the target side.

While the system should learn to also use dictionary entries it has not seen during training, the system needs enough examples in order to learn how to use dictionary entries in general. Since we are concentrating on very rare words, the number of dictionary entries in the parallel data is relatively small. For larger corpora, we therefore explore whether it is helpful to annotate additional phrases. This was done by also extracting phrases that occur more often (*add. Annot*). However, we did use the same split and also evaluated our approach only on the rare phrases.

3.2 Word representations

A second challenge when building a machine translation system for the targeted scenario is the generation of the correct inflected word form. Since we have seen the new words only in the dictionary, we will often need to generate different inflected word forms that we have neither seen in the dictionary nor in the corpus.

While there have been attempts to generate unknown inflected word forms for dictionary entries (e.g. Niehues and Waibel (2011)) prior to neural machine translation, the ability to represent parts of the words in neural machine translation offer a unique opportunity to model morphological inflection. Therefore, in this work, we concentrated on the word representation used in the NMT system. Thereby, we always use the same representation for the source and the target language. The most commonly used word representation used in state-of-the-art neural machine translation systems are byte-pair-encodings (BPE) (Sennrich et al., 2016b). A second successful approach to represent words in a neural machine translation system are character-based representations, where each word is split into its characters.

While there have been several works on comparing these two representations(e.g. Sennrich (2017), they are mostly concentrating on generating the overall best translation performance. However, in this work, we will focus on the rare words. Since only for these words we need to learn how to generate different inflected forms. For the more frequent words, this is often not that important since all word forms occur several times in the corpus.

Besides the generation of unknown inflected forms, the word representation is also important when learning to copy the annotations to the target. If we look at the example dictionary entry *concentric* $\rightarrow$ *konzentrisch*, the lemma *konzentrisch* got split into the subwords *konzent@@ ris@@ ch* while the inflect form *konzentrischer* into *kon@@ zentr@@ ischer*. In this case there is no overlap in the subwords between the lemma and the inflected form. Therefore, it is difficult for the system to learn from the suggested translation. In contrast, when looking at the character-based representation, the model can copy the lemma and only has to learn to add additional tokens at the end.

In a first step, we compared character-based and sub-word based models. Thereby, we highlight their ability to generate new inflected forms of rare words. For both we used exactly the same NMT architecture. The only difference is that the input and output length for the character-based models is significantly larger since the number of characters is higher than the number of subwords.

We will see that the character-based models are significantly better in generating the different inflected forms for rare words. However, a major challenge is the training time. Due to the significant longer sequence length, also the training and decoding time is much slower. Therefore, we also propose a combination of word-based and character-based models.

In the mixed representation, we split each word that occurs less than k times into its characters, while the other words are kept as they are. Since only frequent words are not split into characters, no further subword segmentation for these words is performed. Thereby, we can speed up the processing due to a short sequence length, but still have the ability to learn how to inflect rare words. Some dictionary entries contain phrases with many frequent words. In order to be able to better inflect

these words, in a second approach we in addition split also all words within a dictionary phrase into characters. We refer to this technique as *Mix+Ann*.

4 Experiments

We evaluate the approaches on three different data sizes and on two different language pairs (English-German and English-Czech). Since we are focusing on the generation of different morphological forms, we always use the morphologically rich language as the target language.

4.1 Data

For English-to-German we created two datasets with different sizes. A first series of experiments is run on the TED (Cettolo et al., 2012) corpus. We split the corpus into training, validation and test sets as described in Section 2. In addition, we evaluate the system also on the official test sets *tst2014*, *tst2015* and *2018* and report average metrics for these test sets.

For the second system, we use the Europarl corpus (Koehn, 2005). This corpus is around 10 times bigger than the TED corpus as shown in Table 3. In addition to the target test set, we also tested the systems on the test2006 and test2007, which are the most recent official test sets from the same domain used for the WMT.

Finally, we also tested the techniques on a different language pair. For this we choose English to Czech and also use the Europarl corpus for these experiments. Since there is no official in-domain corpus available, we tested the systems also on the newstest2019 test set.

As shown in Table 3, the parameters mentioned in Section 2 lead to a reasonable test set size for all corpora. As mentioned in Section 3.1, we evaluate the system on Europarl with different amounts of training annotations. All data sets with their splits are available for further experiments [4].

	EN-DE		EN-CS
	TED	Europarl	Europarl
Train	198K	1.9M	636K
- Annot	1.6K	1.2K	2.7K
- add. Annot		14.5K	24.3K
Valid	1610	1196	2000
Test	3181	2140	5360

Table 3: Data size in number of sentences

[4]https://nlp-dke.github.io/data/rareWordNMT/

4.2 System

All data was processed using the Stanza toolkit (Qi et al., 2020) for tokenization and lemmatization. The lemmatization was only used for matching the dictionary entries, the translation systems were built on the inflected words. If BPE is applied, we used a BPE size of 20K. For the mixed representation, words occurring less than $k = 50$ times were represented as individual characters.

We use the standard transformer architecture (Vaswani et al., 2017) and increase the number of layers to eight. The layer size is 512 and the inner size is 2048. Furthermore, we apply word dropout (Gal and Ghahramani, 2016) with $p = 0.1$. We use the same learning rate schedule as in the original work and the implementation presented in (Pham et al., 2019) [5]. All systems were always trained from scratch with random initialization.

4.3 TED

A first series of experiments were performed on the TED task. We evaluated the one-shot learning approach by source sentence annotation as well as the three different word representations described in Section 3.2. In a first step, we evaluated the translation performance using BLEU (mteval-v14.pl) and characTER (Wang et al., 2016) on the continuous learning test set as well as on the official test set (Table 4).

The baseline systems using no one-shot learning do not annotate the source at all and are trained on the standard parallel data. If we take a look at the official test set, we see systems using character-based representation (*Character* and *Mix*) perform slightly better than the subword-based models. This might be due to the fact that the TED training data is rather small. Secondly, the one-shot learning approach has no influence on the translation performance of this test set. This is not surprising, since only 94 phrases in the 4343 sentences of the test sets were annotated. Therefore, we also evaluated our approach on the dedicated continuous-learning test set (*CL test*), created by the new train-test split.

The improvements by character-based representation on the CL test set are even larger. This might be due to the fact that there are more rare words in these sentences and therefore the advantages of the character-based models is stronger. Secondly, in this case, the one-shot approach improvements improve the translation quality. Since the improve-

[5]https://github.com/nlp-dke/NMTGMinor

Represen-tation	One-Shot	CL Test		official Test	
		BLEU ↑	characTER ↓	BLEU ↑	characTER ↓
BPE	No	25.97	44.09	26.17	44.62
Character	No	28.12	42.79	26.57	44.27
Mix	No	27.44	42.79	26.83	44.28
BPE	Annot	26.00	41.74	26.21	44.73
Character	Annot	28.92	40.16	26.72	43.96
Mix	Annot	28.93	40.96	26.8	44.44

Table 4: Translation quality on TED tasks

Representation	One-Shot	Exact match			Lemma match			Morph. Adjustment		
		All	OneS	FewS	All	OneS	FewS	All	OneS	FewS
BPE	No	34	22	53	31	27	62	29	22	43
Character	No	48	40	60	55	47	68	45	43	48
Mix	No	42	35	54	49	40	63	38	34	46
BPE	Annot	48	34	69	62	46	88	33	24	50
Character	Annot	76	74	78	92	91	93	62	61	64
Mix	Annot	75	72	79	92	91	94	59	56	65

Table 5: Rare word accuracy on TED tasks

ments for the BPE-based system are only measured by characTER and not by BLEU might indicate that for this system it is more challenging to generate the correct inflected form.

To better analyse this, we also perform a detailed evaluation as described in Section 2.3 and shown in Table 5. First of all, the experiments show the difficulty of the task. The baseline system is only able to translate 34% of the phrases correctly. For the one-shot subset this even drops to 22%.

Secondly, the experiments show that the challenge can only successfully be addressed by modelling both: one-shot learning and word representation. On the last two lines using character-based word representation and one-shot learning are able to achieve high accuracy. We see an improvement by 50% percent absolutely, which is a relative improvement by more than 300%. Furthermore, for these models there is no longer a clear difference between the one-shot and few-shot examples (Comparison of Columns *OneS* and *FewS*).

By looking at them separately, we see that only using one-shot learning improves the quality slightly. However, even when ignoring the word infection, the model often is not able to produce the correct lemma. The example in Section 3.2, motivates one challenge when learning to copy with different subword segmentations. If we only use character-based representations, we see improvements, especially for phrases that do not occur in training. In this case, the model is more often able to find the correct translation based on translations of other words. However, a similar performance between the few-shot and one-shot learning is only achieved by combining both techniques.

Finally, when only looking at the words where the lemma is different from the inflected form, we still see open research challenges. While we also could improve the accuracy from around 20% or 30% to nearly 60%, it is still the most difficult case.

While there is no clear difference between the character-based model and the mixed model on the output quality, there is a clear difference in training speed. For the full training on 64 epochs, the character-based model needs 14h, while the mixed representation only needs around 4h. While this is still slower than the subword-based model (2.5h), it still allows for a fast training of the model. Therefore, we only compared the mixed and the sub-word based representation for the remaining experiments on larger corpora.

4.4 Europarl

In a second set of experiments, we evaluated the approach on the larger data set on two different language pairs. In addition to the two word representation from the last experiment (*BPE* and *Mix*), we also applied *Mix+Ann*, where we also represent

Lang.	Representation	One-Shot	CL Test		official Test	
			BLEU ↑	characTER ↓	BLEU ↑	characTER ↓
Ger.	BPE	No	28.74	47.30	25.30	48.75
	Mix	No	30.83	45.80	25.52	48.48
	BPE	Annot	28.74	47.47	25.49	48.64
	Mix	Annot	31.63	44.64	25.45	48.60
	BPE	add.Annot	28.81	47.24	25.45	48.71
	Mix	add.Annot	31.44	44.75	25.50	48.57
	Mix+Ann	add.Annot	31.76	44.11	25.64	48.41
Cz	BPE	No	34.25	39.43	16.2	57.15
	BPE	Annot	34.73	38.39	15.57	57.59
	Mix+Ann	Annot	34.86	38.16	16.62	57.70
	BPE	add.Annot	34.74	38.89	15.7	57.37
	Mix+Ann	add.Annot	35.21	37.95	16.63	57.65

Lang.	Representation	One-Shot	Exact match			Lemma match			Morph. Adjustment		
			All	OneS	FewS	All	OneS	FewS	All	OneS	FewS
Ger.	BPE	No	32	28	42	39	33	50	28	23	37
	Mix	No	42	38	48	50	38	58	37	34	43
	BPE	Annot	47	40	61	61	52	80	35	39	47
	Mix	Annot	66	65	68	83	81	88	51	47	56
	BPE	add.Annot	51	49	55	65	62	70	37	36	38
	Mix	add.Annot	65	63	69	81	78	88	51	40	56
	Mix+Ann	add.Annot	72	72	72	92	91	94	58	56	60
Cz	BPE	No	34	25	53	44	32	67	33	24	51
	BPE	Annot	46	33	70	63	48	92	42	30	67
	Mix+Ann	Annot	64	61	69	92	91	95	60	58	65
	BPE	add.Annot	45	31	71	61	45	91	41	29	58
	Mix+Ann	add.Annot	66	63	72	92	89	95	63	60	70

Table 6: Translation Performance on the Europarl data set

all words within dictionary entries as characters as described in Section 3.2. Furthermore, we also investigate *add. Annot*, where additional dictionary entries were used for more training examples. The results are shown in Table 6.

The overall picture for these experiments and the previous experiments is quite similar. For all three scenarios, the quality of the various systems on the official test sets is relatively similar, however the systems differ when looking especially at the accuracy of translating the dictionary entries. Only when combining one-shot learning with character-based representation, we are able to successfully translate the dictionary entries. Independent of the language pair and data size, we are able to achieve an accuracy of around 70% and an accuracy of around 90% when only looking at the lemmas only. Furthermore, the model performs as good in one-shot learning as in few-shot learning.

However, beside the evidence that the approach works on various language pairs and data sizes, the additional experiments give some more insights. First, although the data is larger, we do not see a difference between the models using additional annotation and the models using only the baseline annotation. So it seems to be sufficient to have around 1000 examples in order to learn to copy the suggestions from the source sentence.

Furthermore, although there are no longer clear improvements for character-based representation on the overall translation performance, also for this experiment with larger data size these representations are essential for the dictionary integration. This is highlighted by the improvements of using characters for all words in dictionary entries (*Mix+Ann*) instead of only for rare words (*Mix*).

5 Related work

In recent years, several different approaches to integrate additional data into neural machine translation have been suggested. If this is parallel data, fine-tuning on the additional, better matching data (Luong and Manning, 2015; Lavergne et al., 2011) is often successful. If the additional data is provided in other forms, different techniques have been investigated.

For human feedback, Turchi et al. (2017) suggested to use fine-tuning on the human generated post edits. Pham et al. (2018) used phrase pairs extracted by statistical machine translation to annotate translations of rare phrases. In the similar scenario Li et al. (2019) used a neural network to store the external phrase pairs.

Even more work has been done to integrate dictionaries into neural machine translation. A first work by Arthur et al. (2016) used the additional dictionary to influence the softmax probabilities of the neural machine translation. Another possibility is to include the dictionary as an additional knowledge source during training using posterior regularization (Zhang et al., 2017). A different approach is chosen by Zhang and Zong (2016) using the dictionary as additional training sentences or generating synthetic sentences. In contrast to this work, these do not allow the integration of new words after training the NMT system.

Several authors investigate the integration of the dictionary as an additional constraint during the coding process (Chatterjee et al., 2017; Hokamp and Liu, 2017; Hasler et al., 2018). This leads to a larger complexity in decoding that has been addressed by Post and Vilar (2018). However, the dictionary is typically a hard constraint which makes it difficult to learn words forms that do not occur in the dictionary.

Most similar to this work is the approach by Dinu et al. (2019), which like this work and Pham et al. (2018) annotates the source sentence with possible translations. They showed that state-of-the-art models no longer need architecture changes, but can directly learn to copy form the source sentences. In this work, we additionally focus on generating new word morphological forms not occurring in the dictionary. We investigated different word representations and analysed their influence on the ability to copy the dictionary entries.

6 Conclusion

By introducing the new continuous learning test set using a different train-test split for existing corpora we could highlight the challenges of state-of-the-art neural machine translation systems. While they achieve very good performance, they are still challenged by new emerging terms. The baseline system was only able to correctly translate 20 to 30 percent of these phrases.

Our integration of bilingual dictionaries into the systems improves the translation performance to correctly translate the words by up to 70%. In 90% of the cases at least the lemma of the word is predicted correctly. Furthermore, in this case, we see no difference in accuracy between words only seen in the dictionary and words also seen a few times in the parallel data. However this is only possible by modelling both: enabling the model to perform one-shot learning and modeling the different morphological forms of the rare phrases. The first one is addressed by annotating the source sentence with dictionary translation while the second one is addressed by using character-based models. By combining character-based and word-based representations we are able to model the different morphological variants of a word as well as enabling the system for fast training.

As mentioned before, this work concentrates on the morphological variants of the dictionary entries and ignores ambiguities due to different possible translation. In the future, we intend to address this by including word sense disambiguation into the translation process.

References

Philip Arthur, Graham Neubig, and Satoshi Nakamura. 2016. Incorporating Discrete Translation Lexicons into Neural Machine Translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1557–1567, Austin, Texas. Association for Computational Linguistics.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 Conference on Machine Translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

M. Cettolo, C. Girardi, and M. Federico. 2012. WIT 3 :

Web Inventory of Transcribed and Translated Talks. In *Proceedings of the 16th EAMT Conference*, pages 261–268, Trento, Italy.

Rajen Chatterjee, Matteo Negri, Marco Turchi, Marcello Federico, Lucia Specia, and Frédéric Blain. 2017. Guiding Neural Machine Translation Decoding with External Knowledge. In *Proceedings of the Second Conference on Machine Translation*, pages 157–168, Copenhagen, Denmark. Association for Computational Linguistics.

Georgiana Dinu, Prashant Mathur, Marcello Federico, and Yaser Al-Onaizan. 2019. Training Neural Machine Translation to Apply Terminology Constraints. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3063–3068, Florence, Italy. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 1027–1035, Barcelona, Spain. Curran Associates Inc.

Eva Hasler, Adrià de Gispert, Gonzalo Iglesias, and Bill Byrne. 2018. Neural Machine Translation Decoding with Terminology Constraints. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 506–512, New Orleans, Louisiana. Association for Computational Linguistics.

Chris Hokamp and Qun Liu. 2017. Lexically Constrained Decoding for Sequence Generation Using Grid Beam Search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1546, Vancouver, Canada. Association for Computational Linguistics.

Catherine Kobus, Josep Crego, and Jean Senellart. 2017. Domain Control for Neural Machine Translation. In *Proceddings of Recent Advances in Natural Language Processing (RANLP 2017)*, pages 372–378, Varna, Bulgaria.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings The Tenth Machine Translation Summit (MT Summit X)*, Phuket, Thailand.

Thomas Lavergne, Hai-Son Le, Alexandre Allauzen, and François Yvon. 2011. LIMSI's experiments in domain adaptation for IWSLT11. In *Proceedings of the International Workshop on Spoken Langugage Translation*, pages 62–67, San Francisco, CA.

Ya Li, Xinyu Liu, Dan Liu, Xueqiang Zhang, and J. Liu. 2019. Learning Efficient Lexically-Constrained Neural Machine Translation with External Memory. *ArXiv*, abs/1901.11344.

Minh-Thang Luong and Christopher D. Manning. 2015. Stanford Neural Machine Translation Systems for Spoken Language Domains. In *Proceedings of the Twelfth International Workshop on Spoken Language Translation (IWSLT 2015)*, Da Nang, Vietnam.

Jan Niehues, Eunah Cho, Thanh Le Ha, and Alex Waibel. 2016. Pre-translation for neural machine translation. In *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers*, pages 1828–1836, Osaka, Japan.

Jan Niehues and Alex Waibel. 2011. Using Wikipedia to translate domain-specific terms in SMT. In *Proceedings of the 8th International Workshop on Spoken Language Translation (IWSLT 2011)*, pages 230–237.

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.

Ngoc-Quan Pham, Thai-Son Nguyen, Jan Niehues, Markus Müller, Sebastian Stüker, and Alexander Waibel. 2019. Very Deep Self-Attention Networks for End-to-End Speech Recognition. In *Proceedings of the 20th Annual Conference of the International Speech Communication Association (Interspeech 2019)*, Graz, Austria.

Ngoc-Quan Pham, Jan Niehues, and Alex Waibel. 2018. Towards one-shot learning for rare-word translation with external experts. In *Proceedings of the Second Workshop on Neural Machine Translation*, Melbourne, Australia.

Matt Post and David Vilar. 2018. Fast Lexically Constrained Decoding with Dynamic Beam Allocation for Neural Machine Translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324, New Orleans, Louisiana. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Rico Sennrich. 2017. How Grammatical is Character-level Neural Machine Translation? Assessing MT Quality with Contrastive Translation Pairs. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 376–382, Valencia, Spain. Association for Computational Linguistics.

Rico Sennrich, Alexandra Birch, and Barry Haddow. 2016a. Controlling Politeness in Neural Machine Translation via Side Constraints. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016)*, pages 35–40, San Diego, California, USA.

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

Marco Turchi, Matteo Negri, Amin Farajian, and Marcello Federico. 2017. Continuous Learning from Human Post-Edits for Neural Machine Translation. *The Prague Bulletin of Mathematical Linguistics*, 108.

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.

Weiyue Wang, Jan-Thorsten Peter, Hendrik Rosendahl, and Hermann Ney. 2016. CharacTer: Translation Edit Rate on Character Level. In *Proceedings of the First Conference on Statistical Machine Translation (WMT 2016)*, pages 505–510, Berlin, Germany.

Jiacheng Zhang, Yang Liu, Huanbo Luan, Jingfang Xu, and Maosong Sun. 2017. Prior Knowledge Integration for Neural Machine Translation using Posterior Regularization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1514–1523, Vancouver, Canada. Association for Computational Linguistics.

Jiajun Zhang and Chengqing Zong. 2016. Bridging Neural Machine Translation and Bilingual Dictionaries.