Abstract

Despite advances in neural machine translation (NMT) quality, rare words continue to be problematic. For humans, the solution to the rare-word problem has long been dictionaries, but dictionaries cannot be straightforwardly incorporated into NMT. In this paper, we describe a new method for "attaching" dictionary definitions to rare words so that the network can learn the best way to use them. We demonstrate improvements of up to 1.8 BLEU using bilingual dictionaries.

1 Introduction

Despite its successes, neural machine translation (NMT) still has unresolved problems. Among them is the problem of rare words, which are paradoxically very common because of Zipf’s Law. In part, this is a problem intrinsic to data-driven machine translation because the system will inevitably encounter words not seen in the training data. In part, however, NMT systems seem particularly challenged by rare words compared with older statistical models.

One reason is that NMT systems have a fixed-size vocabulary, typically 10k–100k words; words outside this vocabulary are represented using a special symbol like UNK. Byte pair encoding (BPE) breaks rare words into smaller, more frequent subwords, at least allowing NMT to see them instead of UNK (Sennrich et al., 2016). But this by no means solves the problem; even with subwords, NMT seems to have difficulty learning translations of very rare words, possibly an instance of catastrophic forgetting (McCloskey and Cohen, 1989).

Humans deal with rare words by looking them up in a dictionary, and the idea of using dictionaries to assist machine translation is extremely old. From a statistical perspective, dictionaries are a useful complement to running text because the uniform distribution of dictionary headwords can smooth out the long-tailed distribution of running text. In pre-neural statistical machine translation systems, the typical way to incorporate bilingual dictionaries is to include them as parallel sentences in the training data. But (as we show), this does not work well for NMT systems.

We are aware of only a few previous attempts to find better ways to incorporate bilingual dictionaries in NMT. Some methods use dictionaries to synthesize new training examples (Zhang and Zong, 2016; Qi et al., 2018; Hämäläinen and Alnajar, 2019). Arthur et al. (2016) extend the model to encourage it to generate translations from the (automatically extracted) dictionary. Post and Vilar (2018) constrain the decoder to generate translations from the dictionary. These approaches all treat dictionary definitions as target-language text, when, in fact, they often have properties very different from ordinary text. For example, CEDICT defines 此致 (cǐzhì) as “(used at the end of a letter to introduce a polite salutation)” which cannot be used as a translation.

In this paper, we present an extension of the Transformer (Vaswani et al., 2017) that “attaches” the dictionary definitions of rare words to their occurrences in source sentences. We introduce new position encodings to represent the nonlinear structure of a source sentence with its attachments. Then the unmodified translation model can learn how to use this attached information. We show that this additional information yields improvements in translation accuracy of up to 1.8 BLEU.

2 Methods

Our method is built on top of the Transformer (Vaswani et al., 2017). For each unknown source word with a dictionary entry, we attach the first 50 tokens of the definition (discarding the rest) to the source sentence. As described below, we encode the definition so as to differentiate it from the source sentence proper and to record which source
word the definition is attached to. We leave the task of deciding whether and how to use the definition up to the translation model, which we use without modifications.

2.1 Position encodings

To differentiate the attached definitions from the source sentence itself, we use special position encodings. An ordinary word \( f \) at position \( p \) is encoded, as usual, as \( \text{E}(f) = \text{WE}(f) + \text{PE}(p) \), where \( \text{WE} \) is the word embedding and \( \text{PE} \) is the usual sinusoidal position encoding (Vaswani et al., 2017).

Suppose that word \( f \) at position \( p \) has an attached definition. Then word \( d \) at position \( q \) of the definition is encoded as

\[
\text{E}(d) = \text{WE}(f) + \text{PE}(p) + \text{WE}(d) + \text{DPE}(q),
\]

where DPE is a position encoding scheme different from \( \text{PE} \). We experimented with several schemes for DPE; in the experiments below, we learned a different encoding for each position (Gehring et al., 2017).

See Figure 1 for an illustration of the encoding of an example source sentence. Note that once all words have received their position encodings, their order does not matter, as the Transformer encoder is order-independent.

2.2 Subword segmentation

To apply our method to data segmented using BPE, we face two new problems. First, since very few words are replaced with \( \text{UNK} \), it is not sufficient to attach definitions to \( \text{UNK} \). How do we decide which words to attach definitions to? Second, if a word has been split into multiple subwords, the definition does not have a single attachment position. How do we represent the attachment position when encoding the definition?

To choose which words to define, we use a simple frequency threshold. If the frequency of a word is above the threshold, we do not attach any definitions. If it is at or below the threshold, we attach the definitions to the first subword. For example, in the sentence in Figure 2, only \( \text{死海} \) (\( \text{sǐhǎi} \)) is at or below the frequency threshold (here, 25), so we attach the definition of \( \text{死海} \) to its first subword, \( \text{死}@@ \).

Figure 1: Our method attaches dictionary definitions to rare words. Here, the source sentence is 大家都知道死海 正在死亡 (\( \text{dàjiā dōu zhīdào sǐhǎi zhèngzài sǐwáng} \), Everyone knows that the Dead Sea is dying). \( \text{WE}(f) \) is the embedding of word \( f \), \( \text{PE}(p) \) is the encoding of position \( p \), and \( \text{DPE}(q) \) is the encoding of position \( q \) within a dictionary definition. The rare word 死海 (\( \text{sǐhǎi} \)) is replaced with \( \text{UNK} \) and defined as the Dead Sea. The words of the definition are encoded with both the position of the defined word (4) and their positions within the definition.

Figure 2: After BPE is applied to the sentence only the rare word 死海 is broken up into 死@@ and 海 thus we attach the definition to 死@@ also at position 4.
| Language  | Task      | train | dev  | test  | total | tokens | types | vocab |
|----------|-----------|-------|------|-------|-------|--------|-------|-------|
| Chi-Eng  | Spoken    | 176,000 | 22,000 | 22,000 | 220,000 | 5.9M | 179k | 22k   |
|          | Science   | 216,000 | 27,000 | 27,000 | 270,000 | 10.1M | 383k | 28k   |
|          | Laws      | 176,000 | 22,000 | 22,000 | 220,000 | 17.4M | 98k  | 22k   |
|          | News      | 360,000 | 45,000 | 45,000 | 450,000 | 23.3M | 477k | 24k   |
|          | Education | 360,000 | 45,000 | 45,000 | 450,000 | 18.6M | 461k | 28k   |
|          | Subtitles | 240,000 | 30,000 | 30,000 | 300,000 | 6.6M  | 147k | 27k   |
|          | Thesis    | 240,000 | 30,000 | 30,000 | 300,000 | 17.2M | 613k | 27k   |
|          | UM-all    | 1,993,500 | 221,500 | 5,000 | 2,2M | 101.3M | 1.3M | 33k   |
| Deu-Eng  | Europarl-small | 160,000 | 20,000 | 20,000 | 200,000 | 10.9M | 151k | 16k   |
|          | Europarl-all | 1,440,000 | 180,000 | 197,758 | 1.8M | 98.6M | 475k | 16k   |

Table 1: Statistics of the various tasks we experimented on. Train/dev/test: number of lines selected for use as training, development, and test data (respectively). Toks: number of word tokens (source+target). Types: number of word types (source+target). Vocab: joint vocabulary size used in word-based experiments.

| Task  | Baseline | Append | Attach |
|-------|----------|--------|--------|
|       | BLEU     | MacroF1| BLEU   | MacroF1| BLEU    | MacroF1|
| Spoken| 14.0     | 13.0   | 13.0   | 12.0   | 14.6    | 13.7   |
| Science| 8.2      | 5.6    | 8.7    | 5.6*e  | 9.2     | 6.0    |
| Laws  | 30.3     | 11.5   | 28.0   | 10.6   | 29.6    | 11.8   |
| News  | 10.8     | 4.4    | 10.4   | 4.3    | 11.4    | 4.9    |
| Education| 8.8    | 5.7    | 8.8*e  | 5.5    | 9.6     | 6.2    |
| Subtitles | 18.8   | 15.5   | 16.5   | 12.8   | 19.4    | 16.4   |
| Thesis | 10.0     | 4.3    | 9.7    | 4.1    | 10.5    | 4.5    |
| UM-all| 16.5     | 18.8   | 17.1   | 18.9*e | 17.3    | 19.9   |
| Europarl-small | 28.3 | 15.8 | 28.0 | 15.4 | 29.0 | 16.8 |
| Europarl-all | 29.1 | 7.5  | 29.0 | 7.5*e | 30.0 | 8.0  |

Table 2: Results on word-based translation. Our method (Attach) significantly improves over the baseline in all tasks other than Laws. By constrast, appending the dictionary to the parallel data (Append) performs worse in most tasks. Differences to the baselines are significant for all tasks except where marked with =. The highest BLEU score and the highest MacroF1 score in each row are written in boldface.

| Task  | Baseline | Append | Attach |
|-------|----------|--------|--------|
|       | BLEU     | MacroF1| BLEU   | MacroF1| BLEU    | MacroF1|
| Spoken| 17.3     | 15.3   | 15.4   | 13.1   | 17.6    | 17.8   | 25     |
| Science| 13.4     | 18.0   | 12.2   | 13.4   | 15.2    | 22.2   | 10     |
| Laws  | 30.1     | 13.7   | 26.8   | 10.6   | 31.0    | 14.4   | 15     |
| News  | 12.6     | 12.8   | 11.8   | 12.2   | 13.2    | 14.5   | 20     |
| Education| 13.3  | 10.7   | 12.3   | 9.5    | 13.9    | 12.6   | 25     |
| Subtitles | 21.6   | 17.3   | 18.5   | 13.5   | 21.5    | 18.5   | 15     |
| Thesis | 15.6     | 17.2   | 14.9   | 16.3   | 16.1    | 18.0   | 15     |
| UM-all| 20.9     | 22.8   | 20.8*e | 22.9*e | 21.1    | 23.6   | 20     |

Table 3: Results on BPE-based Chinese-English translation. Our method (Attach) improves significantly over the baseline in all tasks except for Subtitles. Appending the dictionary to the parallel data (Append) performs worse. All differences are significant except when marked with =. The highest BLEU score and the highest MacroF1 score in each row are written in boldface.

| Task  | Baseline | Append | Attach | Edit |
|-------|----------|--------|--------|------|
|       | BLEU     | MacroF1| BLEU   | MacroF1| BLEU    | MacroF1|
| Europarl-small | 33.5  | 25.9 | 31.7 | 23.1 | 33.6*e | 26.9* | 15     | 33.8 | 27.5 | 15 |
| Europarl-all | 36.4 | 25.5 | 36.3 | 25.0*e | 36.5 | 25.4*e | 15 | 36.8 | 26.1 | 15 |

Table 4: Results on BPE-based German-English translation. Our method with fuzzy matching significantly improves over the baseline. Differences to the baseline are significant except where marked with =.
2.3 Fuzzy Matching

There are multiple morphologically inflected forms for each headword in the dictionary in many languages. Consequently, we extend our approach to find the closest possible dictionary entry for each rare word.

We try to find the dictionary headword with the lowest normalized Levenshtein distance to the rare word for each rare word. The Levenshtein distance between two strings is the minimum number of insertions, deletions, or replacements needed to transform one string to the other; normalized Levenshtein distance divides the number of edits by the length of the longer string. Thus, identical strings have a distance of 0, and completely different strings have a distance of 1. We multiply the DPEs by \((1 - d)\) when attaching the dictionary definitions, where \(d\) is the normalized Levenshtein similarity.

Since computing Levenshtein distance between the entire vocabulary and the entire dictionary would be prohibitively expensive, we used locality sensitive hashing \((\text{LSH})\) to approximate the search more efficiently. We convert the rare word into character trigrams, then into a vector using Minhash \((\text{LSH})\). We then query for dictionary headwords using LSH with a Jaccard similarity score \((\text{LSH})\) of 0.5 or more.

3 Experiments

In this section, we describe our experiments on Chinese-English and German-English translation, comparing our methods – \textit{Attach}, which uses exact matching, and \textit{Edit}, which uses fuzzy matching – against two baselines. One baseline is the standard Transformer without any dictionary information (which we call \textit{Baseline}). The other baseline is the standard Transformer with the bilingual dictionaries included as parallel sentences in the training data (which we call \textit{Append}).

3.1 Data: Chinese-English

For Chinese-English, we used the UM-Corpus \(^3\) \((\text{Tian et al., 2014})\), which has about 2\(\times\)10\(^6\) sentence pairs in eight different domains. Since rare words may be more frequent in certain domains, testing our model on different data types may highlight the conditions where dictionaries can be helpful. We excluded the Microblog domain because of its length (only 5000 lines). For each of the other domains, we split the data into three parts: the first 80\% for training \((\text{train})\), the next 10\% for development \((\text{dev})\), and the last 10\% for testing \((\text{test})\). The task \textit{UM-all} combines all eight domains. The UM-Corpus provides a test set, which we used \((\text{test})\), and we split the provided training data into two parts, the first 90\% for training \((\text{train})\) and last 10\% for development \((\text{dev})\). The exact line counts and other statistics are shown in Table 1.

We used the Stanford segmenter\(^4\) \((\text{Chang et al., 2008})\) for the Chinese data and the Moses tokenizer\(^5\) for the English data.

As a dictionary, we used CC-CEDICT,\(^6\) which has 116,493 entries. Each entry has a traditional Chinese headword (which we delete), a simplified Chinese headword, a pronunciation (which we delete), and one or more definitions. We process the definitions as follows:

\begin{itemize}
  \item Remove substrings of the form \textit{abbr. for} \(c\), where \(c\) is a Chinese word.
  \item If a definition contains \textit{see} \(c\) or \textit{see also} \(c\), where \(c\) is a Chinese word, replace it with the definition of \(c\).
  \item Remove everything in parentheses.
  \item Remove duplicate definitions.
  \item If the entry has no definitions left, delete the whole entry.
  \item Concatenate all the definitions into a single string.
\end{itemize}

The resulting dictionary has 102,567 entries, each consisting of a Chinese headword and a single English definition. We segmented/tokenized these in the same way as the parallel data. The average definition length is five, and the maximum definition length is 107.

For example, consider the following CEDICT entries, where we have already removed traditional Chinese characters and pronunciations for clarity.

\begin{verbatim}
1. Remove substrings of the form abbr. for c, where c is a Chinese word.
2. If a definition contains see c or see also c, where c is a Chinese word, replace it with the definition of c.
3. Remove everything in parentheses.
4. Remove duplicate definitions.
5. If the entry has no definitions left, delete the whole entry.
6. Concatenate all the definitions into a single string.
\end{verbatim}
### Source
- 不只是科学家们对对称性(UNK)感兴趣。
- 我哥哥听说我们做了火药(UNK)。
- 有些登山者经过他身旁，打量(UNK)了他一番

### Definitions
- 对称性: symmetry
- 火药: gunpowder(UNK)
- 打量: to size sb(UNK) up to look sb(UNK) up and down to take the measure of to suppose to reckon

### Reference
- But it’s not just scientists who are interested in symmetry.
- Well, my brother heard that we had made gunpowder.
- Some climbers had come by and looked at him,

### Baseline
- not just the scientists are interested in the **UNK**
- My brother had heard that we had done a **UNK**.
- And some of the climbers passed him and **UNK** him.

### Append
- It’s not just about scientists who are interested in **UNK**.
- My brother has heard that we’ve done a lot of work.
- And some of the **UNK** came over and over and over again.

### Attach
- not only scientists are interested in symmetry in symmetry.
- My brother heard that we had done gunpowder.
- Some climbers passed by him and looked at him,

| Source | Definitions | Reference | Baseline | Append | Attach |
| --- | --- | --- | --- | --- | --- |
| 不只是科学家们对对称性(UNK)感兴趣。 | 对称性: symmetry | But it’s not just scientists who are interested in symmetry. | not just the scientists are interested in the **UNK** | It’s not just about scientists who are interested in **UNK**. | not only scientists are interested in symmetry in symmetry. |
| 我哥哥听说我们做了火药(UNK)。 | 火药: gunpowder(UNK) | Well, my brother heard that we had made gunpowder. | My brother had heard that we had done a **UNK**. | My brother has heard that we’ve done a lot of work. | My brother heard that we had done gunpowder. |
| 有些登山者经过他身旁，打量(UNK)了他一番 | 打量: to size sb(UNK) up to look sb(UNK) up and down to take the measure of to suppose to reckon | Some climbers had come by and looked at him, | And some of the climbers passed him and **UNK** him. | And some of the **UNK** came over and over and over again, | Some climbers passed by him and looked at him, |

We used the German-English dictionary from Stardict,\(^8\) which is derived from Freedict\(^9\) and has 81,628 entries. In this dictionary, the headwords have notes in parentheses indicating things like selectional restrictions; we deleted all of these. Unlike with CEDICT, we did not delete any material in definitions, nor did we resolve cross-references, which were very rare. As before, we removed blank entries and merged multiple definitions into a single line. We tokenized both headwords and definitions with the Moses tokenizer. The final dictionary size is 80,737 entries, with an average definition length of 2.9 and a maximum definition length of 88.

For example, the entry:

(Aktien) zusammenlegen to merge (with)

would become

zusammenlegen to merge (with)

### 3.2 Data: German-English

For German-English, we used the Europarl V7 dataset.\(^7\) We tokenized both sides of the data with the Moses tokenizer. Due to the size of the original Europarl dataset and the increased runtime from our method, we ran some experiments on only the first 200k lines of the dataset, denoted in result tables as Europarl-small, while the full Europarl data is called Europarl-all. We split both into three parts: the first roughly 80% for training, the next 10% for development, and the last 10% for testing. Some statistics of the data are shown in Table 1.

\(^7\)http://statmt.org/europarl/

\(^8\)http://download.huzheng.org/freedict.de/

\(^9\)https://freedict.org/

\(^10\)https://github.com/tnq177/witwicky
We set each system’s vocabulary size close to the 16k operations. For word-based translation, the baseline system, with 16k operations. As described in Section 2.2, we append definitions using 1000 samples, with a significance level of 0.05.

3.4 Results: Word-Based
Table 2 shows results on word-based translation. The Append column shows that simply appending the bilingual dictionary to the parallel training data is unhelpful for all tasks, except UM-all. For UM-all, Append does improve BLEU but not MacroF1. By contrast, our method improves accuracy significantly over Baseline and Append across all tasks except Laws. For Laws, our method improves MacroF1 but not BLEU.

3.5 Results: BPE-Based
As described in Section 2.2, we attach definitions only for words whose frequency falls below a threshold. We found that the optimal frequency thresholds vary on different datasets, and there was no direct correlation to corpus size. For each dataset, we trained models using thresholds of $k = 5, 10, 15, 20, 25,$ and 50. We reported the test scores of the models that had the highest BLEU score on the development dataset.

As before, we compared against the two baselines (Baseline and Append). On Chinese-English (Table 3), we only tested our Attach model since Chinese has essentially no morphological inflection. Appending the dictionary to the parallel data did worse than baseline, significantly so on all tasks.

Table 6: Examples from BPE-based systems on the UM-Spoken data. In the first two examples, the baseline system, even with the dictionary in the parallel data (Append), tries to translate the pieces of unknown words separately and incorrectly (e.g., fire, pills, sex). Our model is able to translate the first and third examples correctly as in Table 5, as well as the second example.

| BPE Source | Baseline | Definition | Reference | Attach |
|------------|----------|------------|-----------|--------|
| 1. 不只是科学家们对对称性感兴趣. | 1. Not just scientists are interested in symmetry. | 1. 对称性: symmetry | 1. But it’s not just scientists who are interested in symmetry. | 1. is not just scientists are interested in symmetry. |
| 2. 我哥哥听说我们做了火药. | 2. My brother has heard of the drugs we made. | 2. 火药: gun powder | 2. My brother heard that we had made gunpowder. | 2. My brother heard that we had done a fire. |
| 3. 有些登山者经过他身旁，打量了他一番 | 3. Some climbers had come by and looked at him. | 3. Some climbers pass him by the side, and they took him over, | 3. Some climbers passed his feet, and he took a second, |

and our cleaned dictionaries are available under an open-source license.\footnote{https://github.com/xjz92/Attach_first_bpe} For BPE-based translation, we used joint BPE with 16k operations. For word-based translation, we set each system’s vocabulary size close to the vocabulary size of the corresponding BPE-based system. For example, the Spoken dataset with 16k BPE applied to the training data has 25,168 word types, so we limited the word-based model to 25,000 word types. The vocabulary size we chose for each data set is shown in Table 1.

For all tasks except UM-all and Europarl-all, we trained for 20 epochs, and used the model with the highest dev BLEU to translate the test set. Due to the massive increase in training data on the UM-all and Europarl-all datasets, we only trained for 10 epochs. Otherwise, the settings are the same across all experiments.

We report BLEU (Papineni et al., 2002) and MacroF1 (Gowda et al., 2021) scores of detokenized outputs against raw references. MacroF1 is the F1 score between the number of correct translated word types between a reference and test dataset, giving a clearer picture of a system’s ability to translate rare words. Both scores are computed using Gowda et al.’s fork of SacreBLEU.\footnote{https://github.com/isi-nlp/sacrebleu} We perform significance testing with bootstrap resampling using 1000 samples, with a significance level of 0.05.
except UM-all. By contrast, our model improved over the baseline significantly across all tasks except Subtitles.

On German-English (Table 4), Append did significantly worse on both datasets, whereas our Attach significantly improved BLEU on the full dataset and MacroF1 on the small dataset. Added fuzzy matching (Edit), however, improved both BLEU and MacroF1 significantly on both the smaller and larger datasets.

### 4 Analysis

To further examine how our methods improve translation, we looked at some examples in our UM-Spoken dev set, shown in Table 5 (word-based) and Table 6 (BPE). The (UNK) tag next to dictionary definitions indicates that the word is outside of the system’s vocabulary.

In the first example, 对称性 (duìchēnxìng, symmetry) is unknown to the word-based systems. Adding the definition to the parallel training data (Append) does not help word-based translation because the word remains unknown, whereas our model correctly generates the translation symmetry. With BPE, the word is broken into three pieces, so that the Append system can correctly generate the word symmetry. But the third character (性, xìng) can also mean “sex,” and together with the following character (xìnggài) can mean “sexy.” This explains why the Baseline and Append systems incorrectly adds the words of sex.

In the second example, 火药 (huǒyào, gunpowder) is unknown, and the definition word gunpowder is also unknown. So none of the systems are able to translate this word correctly (though arguably our system’s generation of UNK is preferable). When we switch to BPE, our model generates the correct translation. The other systems fail because this word splits into two very common words, 火 (huǒ, fire), and 药 (yào, drug), which the system tries to translate separately.

The third example shows what happens when we have a long definition that contains useful information, but is not suitable as a direct translation of the unknown word 打量 (dǎliàng). Here we see that our attachment model generates the word looked, apparently by picking out the word look from the definition and inflecting it correctly for the context. No other models were able to generate a word with a similar meaning.

Please see Appendix A for visualizations of the encoder-decoder attention for these three examples.

We also looked at a few examples from the Europarl-small dev set, shown in Table 7 and 8. In the first example, the definition omission was out of vocabulary, so our model was not able to perform any better than the baselines. However, in the BPE systems, our model was able to properly translate Auslassung to omission while none of the other baseline systems was able to.

### 5 Discussion

In Section 1, we mentioned several other methods for using dictionaries in NMT, all of which treat dictionary definitions as target-language text. An alternative approach to handling rare words, which avoids dictionaries altogether, is to use word embeddings trained on large amounts of monolingual
Table 8: Examples from BPE-based systems run on the Europarl-small data. In the first example, unlike in Table 7, the unknown word Auslassung is not replaced with UNK but is split into subwords, which the baseline system as well as the system with the dictionary in its parallel data (Append) translate incorrectly. Our model successfully uses the dictionary definition, omission. In the second example, BPE enables all models to translate the compound Alternativlösung correctly.

data, like fastText embeddings (Bojanowski et al., 2017). Qi et al. (2018) find that fastText embeddings can improve NMT, but there is a sweet spot (likely between 5k and 200k lines) where they have the most impact. They also find that pre-trained embeddings are more effective when the source and target languages are similar.

We, too, experimented with using fastText word embeddings in our NMT system, but have not seen any improvements over the baseline – perhaps because our datasets are somewhat larger than those used by Qi et al. (2018). We also experimented with using dictionaries to improve word embeddings and found that the present approach, which gives the model direct access to dictionary definitions, is far more effective.

The most significant limitation of our method is runtime: because it increases the length of the source sentences, training and decoding take 2–3 times longer. Another limitation is that the effectiveness of this method depends on the quality and coverage of the dictionaries.

In the future, we plan to experiment with additional resources, like thesauruses, gazetteers, or bilingual dictionaries with a different target language. Second, from our examples, we see that our model is able to select a snippet of the definition and adapt it to the target context (for example, by inflecting words), but further analysis is required to understand how much the model is able to do this.

6 Conclusion

In this paper, we presented a simple yet effective way to incorporate dictionaries into a Transformer NMT system, by attaching definitions to source sentences to form a nonlinear structure that the Transformer can learn how to use. We showed that our method can beat baselines significantly, by up to 1.8 BLEU. We also analyzed our system’s outputs and found that our model is learning to select and adapt parts of the definition, which it does not learn to do when the dictionary is simply appended to the training data.

Acknowledgements

This paper is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract #FA8650-17-C-9116. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Philip Arthur, Graham Neubig, and Satoshi Nakamura. 2016. Incorporating discrete translation lexicons into neural machine translation. In Proc. EMNLP, pages 1557–1567.
A Attention Visualizations

Figures 3 and 4 show visualizations of the attention of our Attach model. They show the first layer of encoder-decoder attention when translating the three Chinese sentences of Tables 5 and 6. Note the translations are not exactly the same as shown above, because we used a beam size of one instead of the default of four.
Figure 3: Attention visualizations for the first two Chinese-English examples of Tables 5 and 6.
Figure 4: Attention visualizations for the third Chinese-English example of Tables 5 and 6.