TSMind: Alibaba and Soochow University’s Submission to the WMT22 Translation Suggestion Task

Xin Ge1,∗, Ke Wang1,∗, Jiayi Wang1, Nini Xiao1,2, Xiangyu Duan2, Yu Zhao1, Yuqi Zhang1,†
1Alibaba Group Inc. 2Soochow University
{shiyi.gx, moyu.wk, joanne.wjy}@alibaba-inc.com
{nnxiaonnxiao, xiangyuduan}@suda.edu.cn, {kongyu, chenwei.zyq}@alibaba-inc.com

Abstract

This paper describes the joint submission of Alibaba and Soochow University, TSMind, to the WMT 2022 Shared Task on Translation Suggestion (TS). We participate in the English ↔ German and English ↔ Chinese tasks. Basically, we utilize the model paradigm fine-tuning on the downstream tasks based on large-scale pre-trained models, which has recently achieved great success. We choose FAIR’s WMT19 English ↔ German news translation system and MBART50 for English ↔ Chinese as our pre-trained models. Considering the task’s condition of limited use of training data, we follow the data augmentation strategies proposed by Yang et al. (2021) to boost our TS model performance. The difference is that we further involve the dual conditional cross-entropy model and GPT-2 language model to filter augmented data. The leader board finally shows that our submissions are ranked first in three of four language directions in the Naive TS task of the WMT22 Translation Suggestion task.

1 Introduction

Computer-aided translation (CAT) (Barrachina et al., 2009; Green et al., 2014, 2015; Knowles and Koehn, 2016) has become more and more popular to help increase the quality of machine translation (Lopez, 2008; Koehn, 2009) result. It also improves the efficiency of translators by combining the results of machine translation and the content edited by translators in the process of translation or post-editing (Bowker, 2002; Lengyel and Ugray, 2004; Bowker and Fisher, 2010; Bowker, 2014; Chatterjee, 2019).

Post-editing based on machine translation is typical in CAT. Recent works (Domingo et al., 2016; González-Rubio et al., 2016; Peris et al., 2017) propose interactive protocols and algorithms so that humans and machines can collaborate during translation, and machines can automatically provide feedback on humans’ edits. One interesting mode is Translation Suggestion (TS) (Yang et al., 2021), which offers alternatives for specific spans of words in the generated machine translation. It will be convenient if the model refines translation results in those specified locations with potential translation errors. Yang et al. (2021) released a benchmark dataset for TS, WeTS, which is one of the shared tasks in WMT22. At the same time, they proposed an end-to-end Transformer-like model for TS as the benchmark system.

However, the lack of many labeled TS data limits the training of a large Transformer model to some extent. Though Yang et al. (2021) have tried to utilize XLM-Roberta (Conneau et al., 2019) to initialize the encoder of the Transformer, the decoder has to be trained from scratch, which leads to relatively low BLEU scores for some specific TS spans. We investigate the potential of other encoder-decoder pre-trained models by experiments to see if there is still room for improvement. Finally, we have found that pre-trained Transformer NMT models could be suitable choices to be fine-tuned with the limited size of TS data. In addition, we applied similar data augmentation strategies proposed in Yang et al. (2021), but use the well-trained alignment models between source and target languages from Lu et al. (2020) to filter out high-quality augmented data. Our submissions are ranked first in three of four language directions in the WMT22 Translation Suggestion task.

2 The Model

We train a simple end-to-end Transformer model for each language pair to generate the translation suggestion candidates. The source sentence and the masked translation, in which an incorrect span requiring an alternative has been replaced with a special mask tokens in advance, are concatenated with a special separation token [SEP]. Afterward,
we feed the concatenated sequence as input of the Transformer encoder and the translation suggestion needs to be generated by the Transformer decoder. The model is trained in the same way of a normal translation model.

Considering that the TS task also relies on alignments of hidden representations between the source and the target language, a well-trained translation model can be a good starting point for TS model training. The weights of our model are initialized with a pre-trained Transformer NMT model. Then, a two-phase training pipeline is applied. In the first phase, the model is trained with pseudo corpus derived from data augmentation described in Section 3. In the second phase, we fine-tune the model with the real TS train data released by the organizers.

3 Data Augmentation

We follow the data augmentation methods provided by (Yang et al., 2021) to generate three types of pseudo data for TS model training: data sampled on the golden parallel corpus, data extracted with word alignment. However, the details of the pseudo data augmentation in this paper are slightly different from those of Yang et al. (2021). Full details are exhibited in the following subsections.

![Table 1: Notations](image)

Table 1: Notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>Sentence in source language</td>
</tr>
<tr>
<td>y</td>
<td>Machine translation result of x</td>
</tr>
<tr>
<td>r</td>
<td>Reference sentence x</td>
</tr>
<tr>
<td>x^i</td>
<td>The i-th token of x</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>x_{i:j}</td>
<td>The fragment of x from position i to j</td>
</tr>
<tr>
<td>x_{i:j}^*</td>
<td>The masked version of x, in which tokens at the position from i to j of x is replaced with a mask token.</td>
</tr>
<tr>
<td>p</td>
<td>All aligned-phrase pair between y and r, pair looks like (y_{i:j}, r_{a:b})</td>
</tr>
<tr>
<td>ŷ</td>
<td>Replace y_{i:j} with r_{a:b} in y, and get another new sentence ŷ</td>
</tr>
</tbody>
</table>

![Table 2: Number of parallel samples remained after filtering by length and cross-entropy quality score](image)

Table 2: Number of parallel samples remained after filtering by length and cross-entropy quality score (Lu et al., 2020).

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>WMT22</td>
<td>Filter Length</td>
</tr>
<tr>
<td></td>
<td>Filter Quality</td>
</tr>
<tr>
<td>en-zh</td>
<td>23.2M</td>
</tr>
<tr>
<td>en-de</td>
<td>30.0M</td>
</tr>
</tbody>
</table>

3.1 Sampling from golden parallel corpus

Raw parallel corpus is firstly filtered by the sentence length. All sentence pairs that have less than 20 words or more than 80 words on any side are removed.

Considering that there might be noise data in the corpus, we apply the dual conditional cross-entropy model (Lu et al., 2020) to obtain a quality score for each sample. Sentence pairs with low quality are
filtered.

Then we generate a pseudo corpus with the remained high-quality parallel corpus. \((x, r)\) is marked as the sentence pair of the parallel corpus, where \(x\) is the source sentence and \(r\) is the golden reference. \(\|r\|\) represents the number of tokens in \(r\).

The first step is to randomly sample the length \(l\) to mask for the reference \(r\) from a uniform distribution:

\[
l \sim U(1, \|r\|)
\]

Then a span with \(l\) tokens \(r^{i:j}\) is randomly selected by:

\[
i \sim U(0, \|r\| - l), \quad j = i + l
\]

Finally, we get the TS training data \((x, r^{-i:j}, y^{i:j})\) from each parallel sentence pair \((x, r)\), where \(r^{-i:j}\) is denoted as the masked version of \(r\), in which \(r^{i:j}\) is replaced with a mask token, e.g. <MASK_REP>.

### 3.2 Sampling on Pseudo Parallel Corpus

In addition, the monolingual corpus is another source for data augmentation. We first filter the monolingual data with a language identification process. Then pseudo parallel corpus is generated with NMT models. Finally, TS training data can be generated as we do in Section 3.1.

### 3.3 Extracting with Word Alignment

In the task of TS, the labels for the masked span is always correct while the translation contexts of the span, \(y^{-i:j}\) are not error-free. Therefore, both of the above two types of pseudo data are biased from the task. In pseudo data sampled from golden parallel corpus, the translation contexts are error-free. And the labels of pseudo data from machine translation results are not always correct. To reduce the bias, another way of data augmentation is proposed in Yang et al. (2021). They utilize the alignment between the machine translation and the golden reference to generate pseudo-training samples for TS. We use the similar idea and the details of our alignment-based data augmentation algorithm are described as follows.

Given the triplet \((x, y, r)\) where \(x\) is the source sentence, \(y\) is the machine translation result generated by NMT models, and \(r\) is the reference, we need to find aligned segment pairs \((y^{i:j}, r^{a:b})\) between \(y\) and \(r\).

First, we use the Fast Align toolkit (Dyer et al., 2013) to extract token alignments between \(y\) and \(r\). The align result \(A\) is a list of aligned indexes in the format of \(i:a\), which means token \(y^i\) is aligned to \(r^a\). With the token alignments, the next step is to extract aligned-phrase pairs, denoted as \(p\). Figure 1 shows an example of an aligned phrase between MT and reference. The algorithm of the aligned-

![Figure 1: In this example, we have the alignment info between machine translation (MT) and reference sentences: 0-0, 1-1, 2-2, 3-3, 4-4, 6-6, 6-7, the phrase from 0 ~ 4 in MT are aligned to 0 ~ 4 in reference. The rectangle enclosed by the aligned phrases between MT and reference should satisfy that each row and each column has at least one *.

<table>
<thead>
<tr>
<th>Machine Translation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-0 1-1 2-2 3-3 4-4 6-6 6-7 7-7 ...</td>
<td></td>
</tr>
<tr>
<td>0 1 2 3 4 5 6 7 ...</td>
<td></td>
</tr>
<tr>
<td>All revenue of the system ...</td>
<td></td>
</tr>
<tr>
<td>All revenues from the system ...</td>
<td></td>
</tr>
</tbody>
</table>

...
Table 3: Sacre-BLEU on the validation sets of Sub-Task 1 (Naive TS) of the WMT’22 Translation Suggestion Task.

<table>
<thead>
<tr>
<th>Method</th>
<th>En-De</th>
<th>De-En</th>
<th>En-Zh</th>
<th>Zh-En</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSMInd</td>
<td>45.90</td>
<td>43.37</td>
<td>30.21</td>
<td>28.77</td>
</tr>
<tr>
<td>-w/o first-phase training</td>
<td>37.14</td>
<td>33.23</td>
<td>21.20</td>
<td>16.44</td>
</tr>
<tr>
<td>-w/o second-phase training</td>
<td>37.37</td>
<td>36.83</td>
<td>21.84</td>
<td>19.19</td>
</tr>
</tbody>
</table>

4 Experiment

4.1 Corpus and Setup

Parallel corpora for data augmentation in Section 3.1 and 3.3 and monolingual corpora for Section 3.2 are all downloaded from WMT22 general translation task\(^1\). For English ↔ German, WikiMatrix (Schwenk et al., 2021), News Commentary v16, Common Crawl Corpora, and Tilde MODEL Corpora (Rozis and Skadinš, 2017) are used as parallel corpus. For English ↔ Chinese, parallel corpus we used includes UN Parallel Corpus V1.0 (Ziemski et al., 2016) and all parallel corpora from CCMT corpus (Yang et al., 2019) except for the casict2015 corpora. For monolingual corpora, News Commentary and News Crawl are used for all three languages, and Leipzig Corpora (Goldhahn et al., 2012) is also used for Chinese and German.

\(^1\)https://statmt.org/wmt22/translation-task.html
Then the filtering strategies proposed in Section 3.1 are applied to the raw parallel data. The number of data remained after every filtering step can be found in Table 2.

We download monolingual data from WMT22, and get a total of 45.02 million German, 14.68 million English and 10.01 million Chinese monolingual sentences.

For data augmentation in Sections 3.2 and 3.3, we use the NMT models for English ↔ German and English ↔ Chinese released by Yang et al. (2021)\(^2\) to translate the source sentences. And the hyper-parameter \(\beta_1\) and \(\beta_2\) to filter aligned phrases are set to 2.5 and 0.05, respectively.

### 4.2 Model Training

As mentioned in Section 2, a well-trained NMT model is a good starting point for the TS model. For English ↔ German, we initialize the weights with the NMT models released by Ng et al. (2019) (Winner of WMT’19). For English ↔ Chinese, the one-to-many and many-to-one mBART50 models (Tang et al., 2020) are used.

We use the fairseq toolkit (Ott et al., 2019) to train and evaluate our model. Hyper-parameters are set to the same as examples in the fairseq toolkit except that we reset the learning rate at the beginning of the first phase training and beam size is set as 6 during inference.

### 4.3 Experimental Results

We evaluate the TSMind by calculating the Sacre-BLEU (Post, 2018) of the top-1 generated translation suggestion candidate on the golden reference. Results of the validation sets of WMT22 are shown in Table 3. Without first-phase training, we get much worse performances. This demonstrates that a large amount of pseudo corpora contributes much to the model. However, without the second-phase training (i.e. without the human-labeled data), we cannot obtain a good translation suggestion model with only pseudo corpora either. Therefore, the design of the two-phase training and the pseudo corpora are essential to set good translation suggestions.

Since the development set of WMT’22 is not the same as the test set used in Yang et al. (2021), to make a fair comparison, we also report the Sacre-BLEU on the test set of WeTS in Table 4. Results of all baseline systems are reported by Yang et al. (2021). TSMind outperforms the strong baseline, SA-Transformer, significantly with a gap of 7.04 BLEU on average for all four language pairs. We notice that TSMind does not perform well on the English to Chinese language pair. The reason might be that the pre-trained model we use is the one-to-many model of mBART50, and the multilingual decoder is not well-trained for Chinese. For example, on the English to Chinese news translation test set of WMT’20 (Barrault et al., 2020), mBART50 only achieves a Sacre-BLEU value of 30.79, while the Sacre-BLEU of state-of-the-art is 49.2.

### 5 Conclusion

In this paper, we present our translation suggestion systems, TSMind, for the WMT 2022 Translation Suggestion Task. Different from previous work, we use well-trained NMT models as the pre-trained models and applied a two-phase training strategy.

We explore three data augmentation strategies from previous work and utilize the dual conditional cross-entropy model to filter out low-quality augmented data. The leader board finally shows that our submissions are ranked first in three of four language directions in the Naive TS task of WMT22 Translation Suggestion task.

### References


---

<table>
<thead>
<tr>
<th></th>
<th>En-De</th>
<th>De-En</th>
<th>En-Zh</th>
<th>Zh-En</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>XLM-R</td>
<td>25.12</td>
<td>27.40</td>
<td>32.48</td>
<td>21.25</td>
<td>26.56</td>
</tr>
<tr>
<td>Naïve Transformer</td>
<td>28.15</td>
<td>30.08</td>
<td>35.01</td>
<td>24.20</td>
<td>29.36</td>
</tr>
<tr>
<td>Dual-source Transformer</td>
<td>28.09</td>
<td>30.23</td>
<td>35.10</td>
<td>24.29</td>
<td>29.43</td>
</tr>
<tr>
<td>SA-Transformer</td>
<td>29.48</td>
<td>31.20</td>
<td>36.28</td>
<td>25.51</td>
<td>30.62</td>
</tr>
<tr>
<td>TSMind</td>
<td><strong>47.44</strong></td>
<td><strong>45.02</strong></td>
<td><strong>36.28</strong></td>
<td>31.78</td>
<td><strong>37.66</strong></td>
</tr>
</tbody>
</table>

Table 4: Sacre-BLEU on the test sets of WeTS (Yang et al., 2021)

---

\(^2\)https://github.com/ZhenYangIACAS/WeTS


