YerevaNN's Systems for WMT20 Biomedical Translation Task: The Effect of Fixing Misaligned Sentence Pairs

Karen Hambardzumyan[1], Hovhannes Tamoyan[1,2], and Hrant Khachatrian[1,3]

[1]YerevaNN
[2]American University of Armenia
[3]Department of Informatics and Applied Mathematics, Yerevan State University

Abstract

This report describes YerevaNN's neural machine translation systems and data processing pipelines developed for WMT20 biomedical translation task. We provide systems for English-Russian and English-German language pairs. For the English-Russian pair, our submissions achieve the best BLEU scores, with en→ru direction outperforming the other systems by a significant margin. We explain most of the improvements by our heavy data preprocessing pipeline which attempts to fix poorly aligned sentences in the parallel data.

1 Introduction

Biomedical machine translation is a perfect playground to develop narrow domain neural machine translation models. In such tasks, the available parallel in-domain data is usually limited and noisy which creates many challenges.

In the previous works (Bawden et al., 2019), researchers focused on transfer learning methods (Saunders et al., 2019) or attempted to mix the training data with other sources (Peng et al., 2019) to address the issue of data scarcity. In this work, we show that the transfer performance is very dependent on the quality of the training data, and with a little effort, it is possible to improve the given MEDLINE training data and gain a significant performance boost.

We have manually created a much higher quality subset of the original MEDLINE training data for local evaluation purposes. The insights collected during this manual analysis was then used to fix the most common issues within the training data. In particular, we noticed that the original dataset contained paper abstracts in two languages without sentence-level alignments, and the training corpus provided by the organizers was created using an automated sentence segmentation and alignment process, which was not perfect. We built a data pipeline[1] that handles 1) cleanup, 2) sentence segmentation, 3) alignment of translation sentence pairs and 4) preprocessing.

In our experiments, we did not use any data source other than MEDLINE. We chose our baseline model and two other models with the highest BLEU scores on a local test set as our three submissions. The best ones got 35.2% BLEU on English-German and 41.3% on German-English test sets. For English-Russian and Russian-English directions we reached BLEU scores of 37.9% and 43.2% respectively, which are the best scores among all submissions of WMT20 Biomedical Translation Task. Moreover, our models are cheap to train: the average training time of our best models is approximately 30 minutes on a single NVIDIA Titan V GPU.

The paper is organized as follows: Section 2 presents fine-tuning details and evaluation methods for our NMT systems, Section 3 describes the data used in the experiments and the data processing pipeline, Section 4 presents our novel method of monotonic alignment based on multilingual language models. Section 5 discusses the results.

2 System Description

2.1 Pretrained Models

All our NMT models are built on top of WMT19 News Translation task winner models by Ng et al.. We employ FairSeq library (Ott et al., 2019) to fine-tune pretrained models on the in-domain translation data.

The pretrained models are based on `transformer_wmt_en_de_big` architecture (Vaswani et al., 2017) with a modified feedforward

[1]Our data pipeline is available at https://github.com/YerevaNN/parasite

Proceedings of the 5th Conference on Machine Translation (WMT), pages 820–825
Online, November 19–20, 2020.

dimension (8192) and a shared matrix for input and output embeddings. Additionally, en↔de models share vocabulary and embeddings for both source and target sides.

2.2 Fine-Tuning

We start fine-tuning `single_model` versions of Facebook's WMT19 models[2] on in-domain parallel data and stop the training when the perplexity on the validation set does not improve for 5 consecutive epochs.

To fight noisy training data we use label-smoothed cross-entropy loss (Müller et al., 2019).

The neural architecture and related implementation details cannot be changed in the fine-tuning scenario. While this limited our experimental setup, however, it also allowed us to care less about hyperparameter tuning and focus on other parts of the pipeline.

2.3 Implementation Details

The hyperparameters for our baseline models (`run1`) are as follows. The models are fine-tuned on the training data using an inverse-square-root learning rate schedule with 4000 warm-up steps with an initial learning rate of 10^{-5}. Instead of using a fixed batch size, we make batches of maximum 3584 tokens to fit in the memory. For label smoothing, we set a smoothing coefficient of 0.1. Unlike the pretrained models, we use standard Adam betas and disable dropout.

Training with bigger batches (implemented using gradient accumulation, a single update per 128 batches) not only helped us to reduce total training time 4x but also resulted in better models (including our best submissions `run2` and `run3`).

All the models are trained on a single NVIDIA Titan V GPU with 16-bit floating-point operations. The average duration of fine-tuning with bigger batches was 30 minutes.

Finally, we use a beam size of 32 in the inference mode.

2.4 Evaluation and Model Selection

We use two kinds of validation sets for model selection. For early stopping, we calculate the perplexity on a regular validation set which is extracted from the training data. To determine our best models for submissions, we use a separate in-domain dataset which we call "local test set" and calculate BLEU score on it. All BLEU scores are calculated with SacreBLEU case-insensitive configuration.

3 Data

3.1 Parallel Data

For all directions, we use only MEDLINE training data provided by the shared task organizers. We take random 50 documents from the training data as the validation set. In case of en↔de we use OK-tagged sentence-pairs from WMT'19 biomedical translation test set (Bawden et al., 2019) as the *local test set*. To have a *local test set* of a similar quality for en↔ru, we take another random 50 documents, then manually fix misaligned sentences and filter out a few pairs with incorrect translations.

During the manual review of the en↔ru *local test set* we noticed that the provided data was poorly aligned, and it was possible to get high-quality sentence pairs by re-aligning the sentences (only 9 sentences were dropped except the titles/subtitles, out of 504 sentences). Then we tried to use these insights to build a new automated system for monotonic alignment of the sentences (described in Section 4).

Table 1 exhibits the most common issues found in the MEDLINE training data:

- The bitext documents may be misaligned: the translation of a source sentence may appear on a different line, or even on multiple lines, in the target side,
- Headings and section names may occur next to a sentence on one side only, or on both sides,
- English documents may start with titles (often wrapped in brackets), while the Russian ones do not.

These issues are too common in the training set, and simply removing incorrect pairs of sentences would significantly reduce the dataset. Instead, we decided to fix the misaligned sentences to preserve as much parallel content as possible. The solution is described in Section 4.

3.2 Monolingual Data

Although the base models we use are already trained with backtranslation, we try to fine-tune with backtranslation as well. We obtain translations

[2] `wmt19.en-de.joined-dict.single_model` `wmt19.de-en.joined-dict.single_model`, `wmt19.en-ru.single_model`, `wmt19.ru-en.single_model`

1	*[Risk factors of stroke in men exposed to environmental factors at workplace].* OBJECTIVE	Цель исследования - изучение факторов риска развития инсульта у мужчин разных возрастных групп, подвергающихся воздействию неблагоприятных производственных факторов.
2	To explore risk factors of stroke in men of different age groups exposed to adverse environmental factors at work.	*Материал и методы.*
3	*MATERIAL AND METHODS* Four hundred and eleven men after stroke, aged from 30 to 65 years, including 335 patients, who had been exposed to adverse environmental factors at work, were compared to 76 patients who had not been exposed to adverse environmental factors.	Обследованы 411 мужчин в возрасте от 30 до 65 лет, перенесших инсульт, из них 335 пациентов подвергались влиянию неблагоприятных производственных факторов и 76 пациентов, которые воздействия вредных факторов не испытывали (группа сравнения).
4	*RESULTS*	*Результаты.*
5	The distribution of the frequencies of risk factors of stroke depending on the character of adverse factors was shown.	Установлена частота распределения факторов риска развития инсульта у мужчин в зависимости от характера профессиональных вредностей.

Table 1: A hand picked example from MEDLINE en↔ru training set (document #26978637) which demonstrates the most common issues in the dataset. The first line in English includes the title of the paper which is not present in Russian. The English version of the main content of the first line in Russian is given on the second line. Line 3 in English has an extra heading which corresponds to Line 2 on the right side. The rest of the third line on the left matches to the third line on the right side, and the last two lines are correct.

with the fine-tuned models mentioned above, then fine-tune *new* models on a mixed data consisting of the regular parallel training data and backtranslated data with equal proportions.

To perform backtranslation we need a set of in-domain monolingual sentences that do not overlap with the test set. To train backtranslated de↔en and ru ↔ en directions, we took all English sentences from all parallel corpora available from MEDLINE (both training and test sets) excluding the parallel corpora we would eventually train on. This way we collected 296,052 (236,379) English sentences for German (for Russian). To obtain a parallel corpus we translated them using our models, and then filtered them using the same process as with the regular training data (see the next subsection). We ended up with 281,054 (220,916) sentence pairs for en↔de (en↔ru).

We did not perform backtranslation from Russian or German (directions en→de and en→ru), as we did not expect to find in-domain sentences that are not present in MEDLINE.

When translating the monolingual sentences, we tried `sampling`, `sampling-top5`, `greedy`, `beam`, `beam+noise` decoding methods similar to (Edunov et al., 2018), but no major difference in terms of the final BLEU score has been observed.

3.3 Preprocessing

The preprocessing pipeline for our models has to be identical to the one used for pretrained models. First, we perform punctuation normalization (quotation, commas, numbers, replacing punctuation and removing control characters) using `SacreMoses` library. Then, we tokenize the resulting sentences using Moses (Koehn et al., 2007) tokenizer with aggressive dash splits and escaping XML entities. Finally, we use subword segmentation (Sennrich et al., 2016) (`fastbpe` implementation) with BPE codes from pretrained models, with 24k and 32k splits for Russian and for joint English & German, respectively.

We perform additional filtering of the parallel data before the training: we skip those sentence pairs where 1) source or target sentence has more than 250 subwords and/or 2) the ratio of lengths of the source and target sentences is more than 3/2.

During inference, we truncate sentences to the first 1024 subwords (the number of the positional embeddings).

During our early experiments we noticed several issues with our preprocessing pipeline which we fixed for the later experiments. In particular, we noticed that some `sacremoses` command line flags were broken, and the out-of-the-box inference tool from `FairSeq` did not fully replicate the preprocessing pipeline used for training (punctuation normalization and vocabulary-aware subword segmentation). The original pipeline (called *v1*) was used for our baseline models. The later experiments used the fixed implementations of `sacremoses` and `FairSeq` (denoted by *v2*).

4 Monotonic Alignments

The problems of the training set described in Section 3.1 can be caused by poor 1) XML parsing, 2) sentence segmentation, or 3) monotonic segment alignment method. Here we describe a novel method for monotonic sentence alignment using multilingual language models and discuss the contribution of its hyperparameter choices. Multilingual language models have been previously shown to be effective in parallel data mining (Kvapilíková et al., 2020). We also compare our approach to the baseline data pipeline by the shared task organizers which is based on `Syntok` segmentation system and `GMA` (Melamed, 2001).

Our method of monotonic sentence alignments is as follows: we calculate a similarity matrix of all source-target candidate pairs and decode pairs to maximize the similarity of the resulting sentence pairs. We consider two approaches for the decoding step: greedy and dynamic.

4.1 Similarity Matrix

The similarity matrix is calculated using Euclidean distances of sentence embeddings from a pretrained multilingual language model. We found `xlm-roberta-large` (Conneau et al., 2019) to be the best one. In order to obtain a fixed size vector for each sentence, we simply take the average of the wordpiece embeddings (Cer et al., 2018; Artetxe and Schwenk, 2019).

We also attempt to address some common issues concerning the given MEDLINE abstracts that may harm the quality of the alignments: 1) we remove titles from the English version that are absent in the Russian version, 2) we detect the headings that often get attached to adjacent sentences, 3) we lowercase the text before obtaining embeddings (as the English headings are written in capitals, unlike the Russian ones), 4) we experiment with different sentence segmentation systems such as SciSpacy (Neumann et al., 2019) (in-domain, for English) and Razdel[3] (focused on Russian), 5) we also penalize candidates with source/target length ratios exceeding 2. Additionally, we consider using normalized distances and the margin based approach described in (Artetxe and Schwenk, 2019).

[3] `https://github.com/natasha/razdel`

4.2 Greedy Approach

In greedy approach, we construct the set of correct sentence pairs in an iterative process. Given the similarity matrix, at each step we add the sentence pair with the maximum similarity score. As there is an assumption that the alignments should be monotonic, after each step we exclude all remaining candidate sentence pairs that would break the monotonicity. Our implementation finds at most one target sentence for a source sentence (and vice versa).

Algorithm 1: Greedy decoding

$S_N, T_M \leftarrow$ source and target sentences
$D_{i,j} \leftarrow Sim(S_i, T_j)$
$Res \leftarrow \{\}$
while $|Align| < \min(N, M)$ **do**
  $i, j \leftarrow \arg\max(D)$
  $Align \leftarrow Align \cup \{i, j\}$
  $D_{i..N,0..j}, D_{0..i,j..M} \leftarrow 0$
end
Result: Align

4.3 Dynamic Algorithm

In the dynamic algorithm, we consider maximizing the sum of the similarity scores of the selected sentence pairs according to the given matrix. Our implementation of this approach, unlike the greedy one, can produce sentences consisting of multiple (up to K) segments on each side. To find the mapping with the best total similarity score we use dynamic programming.

5 Results

For WMT20 Biomedical Translation Task we prepared three submissions: `run1` for all directions was the baseline model, while for `run2` and `run3` we chose the best models according to their BLEU score on the local test set at the time of the submission. In `run2` and `run3`, all the models besides de→en of `run2` are trained with our data pipeline and bigger batches. The official BLEU scores on samples with "OK" aligned sentences alongside with our local test set are presented in Table 2.

For de→en of `run2`, backtranslation data was collected with beam search (size of 8), in case of ru→en, we had noise added similar to Edunov et al., and for `run3` we used a simple sampling strategy. Our experiments with backtranslation showed no

BLEU Scores on WMT20 Test / Local Test				
Models	**en→de**	**de→en**	**en→ru**	**ru→en**
run1	35.2 / 34.5	41.3 / 45.4	32.6 / 27.7	NA / 30.7
run2		41.4 / 44.7	39.4 / 31.6	43.3 / 33.0
run3	35.2 / 35.1	41.3 / 45.6	37.9 / 31.8	43.2 / 33.1

Table 2: BLEU scores of our submissions

Algorithm 2: One-to-many (K) dynamic decoding

$S_N, T_M \leftarrow$ source and target sentences
$Best_{N,M} \leftarrow 0$
$Res_{N,M} \leftarrow \{\}$
for $i = 1 \rightarrow N$ **do**
  for $j = 1 \rightarrow M$ **do**
    for $u, v = 1 \rightarrow K$ **do**
      $candidate \leftarrow Best_{i-u,j-v} + Sim(S_{i-u..i}, T_{j-v..j})$
      if $candidate > Best_{i,j}$ **then**
        $Best_{i,j} \leftarrow candidate$
        $Res_{i,j} \leftarrow Res_{i-k,j} \cup \{S_{i-k..i}, T_{j-v..j}\}$
      end
    end
  end
end

significant advantage of any of those compared to the others.

For `run2` and `run3`, we used v2 preprocessing, the sentence splitting was done with `scispacy` (for English and German) and a slightly modified version of `razdel` (for Russian).

After our submissions, we further improved our data pipeline. Table 3 is an empirical analysis of the effect of different components of our data pipeline, as measured by the performance on the final translation task. Each row of the table corresponds to a model trained on the data obtained from a pipeline with certain components enabled. There is no other between the rows, all models are trained by fine-tuning the general domain baseline using our default hyperparameters. We measure the BLEU score on the local test set.

Fixing the issues of the standard preprocessing (v2 vs. v1) gives a significant boost, especially when decoding to Russian (en→ru direction). The effect of training with bigger batch sizes gives only a slight improvement, while the absolute training duration reduces drastically.

Model	**en→ru**	**ru→en**
baseline model	27.7	30.7
+ v2 preprocessing	30.5	31.3
+ train with bigger batches	30.7	31.3
+ greedy alignments	30.1	31.8
+ detect section names	30.7	32.3
+ remove titles	31.3	**32.5**
+ optimize total similarity	30.4	32.2
+ normalize distance matrix	30.8	32.1
+ penalize source/target ratio	31.2	31.5
+ one-to-many (K=3)	**32.2**	32.3

Table 3: The effect of different components of the data processing pipeline. We report BLEU scores on the local test set.

As mentioned previously, there were issues with section names and titles in the provided parsed documents. After addressing these issues, our greedy approach gives better alignments.

The total similarity optimization using dynamic programming is not always better than the greedy method, but the performance improves for en→ru with another +1.1% BLEU score. Overall, the new data pipeline gives an enhancement in NMT performance: +1.6% BLEU for ru→en and a bigger gain of +4.5% BLEU score for en→ru.

Although we observe consistent performance improvement for both directions en↔ru, the effect for en→ru direction is more significant. We could not determine the reason for such assymmetry.

6 Conclusion

This work presents the systems our team developed for English-German and English-Russian language pair tracks of WMT20 Biomedical Translation Task. We achieve the best results on the official test set for English↔Russian language pair, outperforming competitors by a significant margin on English→Russian direction. We show that it is possible to improve the performance of neural machine translation models by simply improving the quality of the in-domain parallel

data. The suggested method for monotonic sentence segment alignment based on pretrained multilingual language models demonstrated promising results. We explored how different components of our data processing pipeline contributed to the quality of the resulting translation systems. In future work, we plan to investigate the applicability of this pipeline to a wider set of language pairs and domains.

Acknowledgments

We would like to thank Adam Bittlingmayer from ModelFront for useful discussions on the quality of parallel corpora. All experiments were performed on Titan V GPUs donated to YerevaNN by NVIDIA.

References

Mikel Artetxe and Holger Schwenk. 2019. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203.

Rachel Bawden, Kevin Bretonnel Cohen, Cristian Grozea, Antonio Jimeno Yepes, Madeleine Kittner, Martin Krallinger, Nancy Mah, Aurelie Neveol, Mariana Neves, Felipe Soares, et al. 2019. Findings of the wmt 2019 biomedical translation shared task: Evaluation for medline abstracts and biomedical terminologies. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 29–53.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder for english. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. *arXiv preprint arXiv:1808.09381*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Ivana Kvapilíková, Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Ondřej Bojar. 2020. Unsupervised multilingual sentence embeddings for parallel corpus mining. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 255–262.

I Dan Melamed. 2001. Geometric approach to mapping bitext correspondence.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. 2019. When does label smoothing help? In *Advances in Neural Information Processing Systems*, pages 4694–4703.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. Scispacy: Fast and robust models for biomedical natural language processing. *arXiv preprint arXiv:1902.07669*.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook fair's wmt19 news translation task submission. *arXiv preprint arXiv:1907.06616*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Wei Peng, Jianfeng Liu, Liangyou Li, and Qun Liu. 2019. Huawei's nmt systems for the wmt 2019 biomedical translation task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 164–168.

Danielle Saunders, Felix Stahlberg, and Bill Byrne. 2019. Ucam biomedical translation at wmt19: Transfer learning multi-domain ensembles. *arXiv preprint arXiv:1906.05786*.

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

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*, pages 5998–6008.