PROMT Systems for WMT21 Terminology Translation Task

Alexander Molchanov, Vladislav Kovalenko & Fedor Bykov
PROMT LLC
17E Uralskaya str. building 3, 199155,
St. Petersburg, Russia
First.Last@promt.ru

Abstract

This paper describes the PROMT submissions for the WMT21 Terminology Translation Task. We participate in two directions: English to French and English to Russian. Our final submissions are MarianNMT-based neural systems. We present two technologies for terminology translation: a modification of the Dinu et al. (2019) soft-constrained approach and our own approach called PROMT Smart Neural Dictionary (SmartND). We achieve good results in both directions.

1 Introduction

The currently state-of-the-art approach of neural machine translation (NMT) does not inherently allow for explicit control over the system's output. That is why terminology translation has always been a problem for NMT systems. There are several approaches to solving this problem. One common paradigm is constrained decoding (Hokamp and Liu, 2017; Anderson et al., 2017; Post and Vilar, 2018), where the terminology matches are presented as hard constraints that the beam search must satisfy. Constrained decoding has its disadvantages: it is computationally expensive and can deteriorate the translation quality (Dinu et al., 2019). Another common approach is the one introduced by (Dinu et al., 2019): the terminological constraints are provided as input to the NMT as additional annotations inline with the source sentence. These can be considered 'soft' constraints, as there is no guarantee that the NMT system will indeed produce an output containing them.

In this paper we describe two approaches to terminology translation. First, we propose a modification of the (Dinu et al., 2019) approach. Second, we introduce our own technology PROMT Smart Neural Dictionary (SmartND) aimed at handling terminology translation.

The paper is organized as follows: in Section 2 we describe the systems built for the Task and the data we used. In Section 3 we describe our technologies for terminology translation. In Section 4 we present and discuss the results. We conclude the paper with discussion for possible future work in Section 5.

2 Systems overview

We submitted two single baseline transformer-based (Vaswani et al., 2017) systems trained with the MarianNMT (Junczys-Dowmunt et al., 2018) toolkit: English-Russian and English-French. We use all parallel data allowed by the organizers. The final systems have the same architecture: we use a shared vocabulary of sizes 16k and 32k for the English-French and English-Russian systems respectively. We use the OpenNMT toolkit (Klein et al., 2017) version of byte pair encoding (BPE) (Sennrich et al., 2016b) for subword segmentation. We use the devsets provided by the organizers as our development sets.

We build intermediate models to obtain back-translations (Sennrich et al., 2016a) for our final systems. We use iterative back-translation for the English-Russian system. The intermediate models are trained using SentencePiece (Kudo and Richardson, 2018) for subword segmentation as we noticed that SentencePiece-based models are more robust in low and middle-resource

Proceedings of the Sixth Conference on Machine Translation (WMT), pages 835–841
November 10–11, 2021. ©2021 Association for Computational Linguistics

conditions. We also tag all our synthetic data with special tokens at the beginning of the source sentences as described in (Caswell et al., 2019). Our final models use three types of synthetic data for training: back-translations, data with terminology and special data with placeholders for processing named entities during translation (see Molchanov, 2019 for details). The models are trained with guided alignment which is used at translation time by our SmartND technology. We obtain alignments using the `fast-align` (Dyer et al., 2013) tool. Both final models were trained for approximately 1.2M steps on two RTX 2080 GPUs.

We also perform fine-tuning for our final systems. There are two reasons for that. First, due to time constraints the initial final systems were trained with only one term in each sentence with terminology markup. After testing these systems we realized that they couldn't handle sentences containing multiple terms. The second reason is that we only processed parallel data and not back-translations for the initial final systems, whereas the 2020 news contain a lot of information about COVID etc. For fine-tuning we processed both the back-translated news and parallel data only using the glossary provided by organizers. This data was mixed with general parallel data.

2.1 Data preparation

There are several stages in our data preparation pipeline. These are mostly common filtering techniques. The statistics for the training data are shown in Table 1. The main stages of the pipeline are:

- Basic filtering
  This includes some simple length-based and source-target length ratio-based heuristics, removing tags, lines with low amount of alphabetic symbols etc. We also remove lines which appear to be emails or web-addresses and duplicates.

- Language identification
  The algorithm is a fairly simple ensemble of three tools: `pycld2`[1], `langid` (Lui and Baldwin, 2012), `langdetect`[2]. We only use pycld2 for large monolingual corpora.

- Bicleaner filtering
  We use the bicleaner (Ramírez-Sánchez et al., 2020) tool to filter parallel data. We discard all sentence pairs with the score threshold <= 0.3.

- Scoring with NMT models
  We finally score all parallel data and back-translations with our intermediate models to discard non-parallel sentence pairs and bad synthetic translations.

2.2 English-French

Due to time constraints and relatively large amounts of training data for the English-French pair we only build one intermediate model. We use all parallel data that we suppose to be of good quality (i.e. all data except the paracrawl, commoncrawl and giga corpora; we also randomly select only 2.5M sentence pairs from the United Nations corpus) and build a joint system trained to translate both from English into French and back. Basic filtering is applied to this data. We use a shared vocabulary of size 8k obtained with SentencePiece. We also tag the source side of the training data with language tokens. The model is trained for approximately 1M steps on two RTX 2080 GPUs. We then use this system to 1) score all parallel data in both directions; 2) translate the monolingual French news corpora into English. We translate the 2020, 2019 and 2018 news corpora. The final model is built using all allowed filtered parallel data, back-translated news and additional synthetic data for terminology markup (see Section 3 for details).

2.3 English-Russian

The English-Russian was a surprise pair announced roughly three weeks before the submission deadline. That is why despite the relatively small amounts of parallel data we only make two iterations of training intermediate systems. We first build an English-Russian system using all parallel data (except commoncrawl which we believe to be of bad quality; basic filtering is applied) including the Edinborough corpus of Russian news translated into English and separate SentencePiece-based vocabularies of

[1] https://pypi.org/project/pycld2/

[2] https://pypi.org/project/langdetect/

size 16k each. As there are approximately 25M parallel sentences pairs, we randomly select 25.5M from the back-translated Russian news corpus. We then use this model to translate the English monolingual 2020 news corpus into Russian. Then we build a Russian-English system with the same vocabularies using all completely filtered parallel data and the obtained English-Russian translations. After that we translate the Russian monolingual news (2020, 2019 and 2018) into English. The final English-Russian model is trained on all filtered parallel data (also scored with the two intermediate systems) and the back-translations of the Russian monolingual news. Despite the fact that it is better to use separate vocabularies for models with different alphabets we use a shared vocabulary because this is necessary for our terminology handling approach.

3 Terminology translation

In this section we describe our two approaches to terminology handling in detail.

3.1 SmartND

Our SmartND systems work on the backbone of the PROMT RBMT technology. The technology doesn't need any specific pretraining or fine-tuning. The entire process can be divided into three steps: dictionary creation, terminology search and output modification.

First off we create a PROMT dictionary in specific format based on the provided glossary. The dictionary is highly optimized for speed and

English-Russian						
	#sent	#sent clean	#tok EN	#tok EN clean	#tok RU	#tok RU clean
News-commentary	331,508	263,674	8,940,220	6,833,693	8,483,220	6,490,441
Paracrawl	5,377,911	3,384,721	122,008,867	72,171,449	100,966,255	63,635,823
UN	23,239,280	12,875,296	61,3108,270	401,818,416	578,849,401	375,889,876
WikiMatrix	1,661,908	896,209	39,460,867	22,136,958	36,102,154	19,909,749
Yandex corpus	1,000,000	770,424	24,685,829	18,849,831	22,613,143	17,341,207
Commoncrawl	878,386	309,378	22,000,613	7,375,812	21,152,629	6,712,507
WikiTitles	1,189,058	195,653	3,403,009	839,231	3,515,590	836,091
Total	33,678,051	18,695,355	833,607,675	530,025,390	771,682,392	490,815,694
English-French						
	#sent	#sent clean	#tok EN	#tok EN clean	#tok FR	#tok FR clean
Europarl	1,915,930	1,387,120	53,588,034	38,751,350	59,215,266	43,076,733
News-commentary	365,510	318,811	94,442,52	8,328,207	11,312,937	10,044,909
UN	25,805,088	15,076,117	681,718,544	457,225,777	790,583,218	535,347,782
Commoncrawl	3,244,152	1,832,936	82,530,944	43,761,355	92,685,758	50,241,069
Giga	22,520,376	11,559,142	685,336,581	304,757,715	826,389,803	362,087,754
Paracrawl	104,351,522	45,673,561	2,274,818,705	961,613,380	2,604,498,787	1,078,787,397
Total	158,202,578	75,847,687	3,787,437,060	1,814,437,784	4,384,685,769	2,079,585,644

Table 1: Statistics for the initial and filtered parallel data in sentences (#sent) and tokens (#tok); 'clean' stands for the final filtered versions of the corpora.

contains POS information for each lexeme along with the complete inflectional paradigm. If a term is present in any of our existing dictionaries, we copy the information from there. In other cases, we try to guess the POS and possible paradigm based on how the term ends. Currently, this system only works with nouns, so we omitted any verbs and adjectives present in the provided terminology glossary, as well as any ambiguous terms that could belong to different parts of speech (like ‘quarantine’). If a term has multiple translations we either choose one (if the translations are interchangeable, like ‘World Health Organization’ – 'Organisation mondiale de la Santé' or 'Organisation Mondiale de la Santé') or omit the term entirely. We also drop any common terms that would likely be translated correctly by our NMT models (like ‘coronavirus’) or, in case of the English-Russian language pair, terms with translations that are incorrect (‘Coronavirus crisis’ – 'коронавирус кризис'). This ensures that SmartND will not interfere with a perfectly valid NMT output. We remove 30 entries from the original glossary.

The translation process is organized as follows. If a term is present in the input text, we search the NMT output for the expected term translation. If that translation is not present in the NMT output, our RBMT systems analyze it and determine the grammatical information (case and number) of the word which our NMT model used to translate the term. We use the word-level alignment provided by the NMT model to find the term-translation pair. Then we can substitute that word for the correct translation taken from the RBMT dictionary using the same case and number. The entire process depends of the NMT model providing good quality word-level alignment. We do not substitute the term translation if the alignment is incomplete in that part of the segment.

3.2 Soft-constrained Terminology Translation

Our second approach is based on (Dinu et al., 2019) with slight modifications. The general idea is quite simple: terminology is identified and tagged on the source side, and each term is appended by its translation (also tagged). The work of (Dinu et al., 2019) and (Bergmanis and Pinnis, 2021) is based on the `Sockeye` (Hieber et al., 2017) toolkit. Each part of the term and its translation is marked with a special source feature. Whereas `MarianNMT` doesn’t support source features (and our systems are `MarianNMT`-based), we propose a ‘trick’ similar to the one described in (Tamchyna et al., 2017). We add special tokens after the term and its translation in the input string. In the first version of our systems we added special tokens after each part of the term and each part of its translation to ‘mimic’ a source feature behavior. But we noticed that the resulting strings are often too long, especially if the source line contains several multi-word terms. So we decided to simplify the algorithm and mark each term with three special tokens which indicate the beginning and end of the term itself and the end of its translation: <term_start>, <term_end> and <term_trans>.

We use the glossary provided for the Task to tag our parallel data. We also use the parallel WikiTitles corpora to create more synthetic data with terminology markup. For the English-Russian pair we use the provided WikiTitles corpus. For the English-French pair we use the `wikipedia-parallel-titles`[3] tool to extract the English-French Wikipedia titles. Note that we only use this corpus to indentify and tag terms in the provided constrained data. We apply the basic filtering to the titles corpora and then randomly select 10k parallel entries for data markup. The English-French glossary remains as is, whereas we generate all possible forms for the translations of the English-Russian glossary using our parser to be able to find them in the parallel data and process more sentences for training.

The data preprocessing is simple: we go through the parallel data line by line and identify the terms (either from the provided glossary or from the WikiTitles) on the source side. If a term is found, we look for any of its translations on the target side. If a translation is found, we tag the term and append the found translation as described above. We obtain about 2.1M sentence pairs for the initial system training and around 0.8M pairs for fine-tuning (using only the provided glossary) for the English-Russian pair, For the English-French pair we have around 0.8 sentence pairs for the initial system and 0.2M pairs for tuning.

[3] https://github.com/clab/wikipedia-parallel-titles

At translation time both glossaries remain as is because we don't use the lemmatized approach, so each term is appended by the initial form of the translation. The motivation for this is that we think that having seen different forms of words and expressions at training time the model can 'guess' that it should transform the initial form to the one necessary in this context (i.e. copy and inflect).

4 Results and discussion

In this Section we present the results on the dev and test sets both in terms of automatic and human evaluation and discuss advantages and drawbacks of our approaches. The results of automatic evaluation on the dev sets according to the tool (Mahfuz ibn Alam et al., 2021) provided by the organizers are presented in Table 2. We can see that both our approaches clearly outperform the baseline according to the terminology-related metrics. As for the BLEU (Papineni et al., 2002) scores, they slightly rise for both approaches which indicates positive results of the application of our approaches. We also present the final results for our submitted systems on the test sets in Table 3. They are generally consistent with the results we obtained on the dev sets.

English-French					
	BLEU	**Exact match**	**Window overlap (2)**	**Window overlap (3)**	**1-TERm**
Intermediate	38.45	0.82	0.255	0.253	0.53
Final	45.44	0.87	0.3	0.29	0.61
Final+Soft	45.86	0.966	**0.314**	**0.309**	0.613
Final+SmartND	45.51	0.922	0.307	0.303	0.613
Final tuned	45.29	0.867	0.297	0.289	0.61
Final tuned+Soft	**46.04**	**0.973**	0.309	0.306	**0.614**
Final tuned+SmartND	45.31	0.87	0.299	0.29	0.611
English-Russian					
	BLEU	**Exact match**	**Window overlap (2)**	**Window overlap (3)**	**1-TERm**
Intermediate	23.92	0.707	0.165	0.163	0.395
Final	27.05	0.84	0.205	0.203	0.439
Final+Soft	26.94	0.86	0.2	0.198	0.44
Final+SmartND	**27.22**	0.867	0.208	0.207	**0.44**
Final tuned	26.75	0.742	0.193	0.19	0.433
Final tuned+Soft	26.9	**0.914**	**0.215**	**0.214**	0.438
Final tuned+SmartND	26.91	0.765	0.195	0.191	0.434

Table 2: Results of the Terminolgy Translation Task on the dev sets.

4.1 Tuned models with SmartND

We observe minor decrease in the exact match scores for the tuned models with the SmartND technology. Surprisingly, our final English-Russian tuned system was ranked last on the test set according to the Exact Match and Window Overlap metrics. We performed human evaluation for these translations. The results show that the exact match scores decrease because of the translation of the term *COVID-19* which is translated as *Covid-19* by the tuned model. This is a perfectly fine translation, but the evaluation metric handles all term translations in case-sensitive mode. The tuned model outperforms the baseline model in all other aspects. This is probably a reason to 1) slightly modify our SmartND algorithm; 2) make the scoring metrics more robust regarding the case aspects.

English-French								
	BLEU	**EM**	**WO (2)**	**WO (3)**	**1-TERm**	**COMET**	**Rank (EM)**	**Rank (COMET)**
Final tuned+Soft	47.69	0.974	0.359	0.352	0.625	0.752	**1-3**	3
Final tuned+SmartND	47.89	0.966	0.357	0.348	0.626	0.746	**1-3**	4-5
Best Scores	49.60	**0.974**	**0.359**	**0.352**	0.632	0.781	**-**	-

English-Russian								
	BLEU	**EM**	**WO (2)**	**WO (3)**	**1-TERm**	**COMET**	**Rank (EM)**	**Rank (COMET)**
Final tuned+Soft	31.06	0.909	0.254	0.255	0.482	0.631	**1**	**1-2**
Final tuned+SmartND	31.92	0.788	0.243	0.241	0.487	0.634	10	**1-2**
Final+SmartND	31.52	0.857	0.251	0.250	0.482	0.624	2-5	3
Best Scores	**31.92**	**0.909**	**0.254**	**0.255**	**0.487**	**0.634**	-	-

Table 3: Final results of the Terminolgy Translation Task on the test sets.
EM stands for *Exact Match*, WO stands for *Window Overlap*. The Best Scores row shows the best scores on the test set for each metric from all participants, the PROMT systems are in bold.

4.2 SmartND and Soft-constrained translation

We compared the two approaches to handling terminology during our experiments. They both have advantages and drawbacks originating from their architecture.

The SmartND technology is more reliable as it almost always produces the right translation given the input from the glossary. However, a noisy glossary is a great problem for SmartND as in this case it needs to be carefully handled and filtered by linguists. The second problem with SmartND is that it sometimes (rarely) produces incorrect translations putting words in the wrong form in the output. This concerns morphologically rich languages, and the reason for it is that it is sometimes hard to parse the output and define the correct form for the term translation.

The soft-constrained approach is more robust to noise in terminology glossaries. The NMT output is more fluent as the system tends to put the terms in the right forms or generate its own translation. However, as we noticed, this technology cannot handle very noisy glossaries or entries either. The soft-constrained systems also require specific training and fine-tuning and data for it, which can be costly.

4.3 General translation quality

We also observe the fact that better baseline models receive better scores according to all metrics. We paid more attention to the English-Russian direction in this task and contributed more work to it. As a result, we obtain generally higher scores on the English-Russian direction compared to the English-French direction according to all metrics.

5 Conclusions and future work

In this paper we presented our submissions for the WMT21 Shared Terminology Translation Task. We show good results in both directions we participate (English-French and English-Russian). We are planning to make more thorough analysis of the results of our work on both the dev and test sets. We are also planning to try the lemmatized approach as described in (Bergmanis and Pinnis, 2021).

References

Md Mahfuz ibn Alam, Antonios Anastasopoulos, Laurent Besacier, James Cross, Matthias Gallé, Philipp Koehn, and Vassilina Nikoulina. 2021. On the Evaluation of Machine Translation for Terminology Consistency. *Computing Research Repository*, arXiv:2106.11891.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2017. Guided open vocabulary image captioning with constrained beam search. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 936–945, Copenhagen, Denmark.

Toms Bergmanis and Marcis Pinnis. 2021. Facilitating terminology translation with target lemma annotations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3105–3111, Online.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged Back-Translation. In *Proceedings of the Fourth Conference on Machine Translation*, pages 53–63, Florence, Italy.

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.

Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A Simple, Fast, and Effective Reparameterization of IBM Model 2. In *Proceedings of NAACL HLT 2013*, pages 644–648, Atlanta, USA.

Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2017. Sockeye: A Toolkit for Neural Machine Translation. *Computing Research Repository*, arXiv:1701.02810.

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.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast Neural Machine Translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, Alexander M. Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. *Computing Research Repository*, arXiv:1701.02810. Version 2.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium.

Marco Lui and Timothy Baldwin. 2012. langid.py: An Off-the-shelf Language Identification Tool. In *Proceedings of ACL 2012, System Demonstrations*, pages 25–30, Jeju, Republic of Korea.

Alexander Molchanov. 2019. PROMT Systems for WMT 2019 Shared Translation Task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 302–307, Florence, Italy.

Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 02)*, pages 311–318, Philadelphia, PA, USA.

Marcis Pinnis, Rihards Krišlauks, Daiga Deksne, and Toms Miks. 2017. Neural Machine Translation for Morphologically Rich Languages with Improved Sub-word Units and Synthetic Data. In *Proceedings of the 20th International Conference of Text, Speech and Dialogue (TSD2017)*, pages 237–245, Prague, Czechia.

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.

Gema Ramírez-Sánchez, Jaume Zaragoza-Bernabeu, Marta Bañón and Sergio Ortiz Rojas. 2020. Bifixer and Bicleaner: two open-source tools to clean your parallel data. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 291–298, Lisboa, Portugal.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, Berlin, Germany.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. 2016b. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pages 35–40, Berlin, Germany.

Aleš Tamchyna, Marion Weller-Di Marco and Alexander Fraser. 2017. Modeling Target-Side Inflection in Neural Machine Translation. In *Proceedings of the Second Conference on Machine Translation*, pages 32–42, Copenhagen, Denmark.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA.