

Comparative Analysis of Neural Translation Models based on Transformers Architecture

Alexander Smirnov¹, Nikolay Teslya¹^a, Nikolay Shilov¹^b, Diethard Frank², Elena Minina²
and Martin Kovacs²

¹*SPIIRAS, SPC RAS, 14th line 39, St. Petersburg, Russia*

²*Festo SE & Co. KG, Rüter Str. 82, Esslingen, Germany*

Keywords: Machine Translation, DNN Translation, Comparison, Training, Transformers, Fine-tuning.

Abstract: While processing customers' feedback for an industrial company, one of the important tasks is the classification of customer inquiries. However, this task can produce a number of difficulties when the text of the message can be composed using a large number of languages. One of the solutions, in this case, is to determine the language of the text and translate it into a base language, for which the classifier will be developed. This paper compares open models for automatic translation of texts. The following models based on the Transformers architecture were selected for comparison: M2M100, mBART, OPUS-MT (Helsinki NLP). A test data set was formed containing texts specific to the subject area. Microsoft Azure Translation was chosen as the reference translation. Translations produced by each model were compared with the reference translation using two metrics: BLEU and METEOR. The possibility of fast fine-tuning of models was also investigated to improve the quality of the translation of texts in the problem area. Among the reviewed models, M2M100 turned out to be the best in terms of translation quality, but it is also the most difficult to fine-tune it.

1 INTRODUCTION

With the increasing power of computers, machine learning has made a significant progress recently and has become an efficient mean for automation in various domains (Cioffi et al. 2020; Usuga Cadavid et al. 2020). One of the areas intensively using machine learning is natural text processing, and machine translation (MT, automatic translation of text from one natural language into another) in particular (Jooste, Haque, and Way 2021; Zhang and Zong 2020).

While statistical MT dominated for decades mainly relies on various count-based models, neural machine translation (NMT) does the translation using a neural network (Stahlberg 2020). With the increasing translation quality such models have been widely used both as third party services (Google 2022; Microsoft 2022) or internally hosted services.

Each of the solutions has some advantages and disadvantages. Whereas usage of third-party

translation services does not require corresponding infrastructure to run relatively heavy neural networks, they do not provide much possibilities for fine-tuning to specific terminology, which might be critical in certain scenarios. As a result, hosting an own instance of a neural translation service (either on own hardware or using MaaS (Machine-as-a-Service) services is the only choice in this case.

The presented type of tasks can be used by international industrial companies to automate the processing of requests from customers. Such companies have offices in many countries selling and supporting products. However, for development tasks in the context of digital business transformation, it is important to collect customer quotations and their centralized analysis within a single NLP platform. To process messages in different languages within this platform, it is proposed to automatically translate them into English using neural MT models.

The paper presents an analysis of different state-of-the-art neural MT models for texts from a specific

^a <https://orcid.org/0000-0003-0619-8620>

^b <https://orcid.org/0000-0002-9264-9127>

domain as well as their abilities for fine tuning based on examples from specific terminology in the area of industrial automation.

The paper is structured as follows. Section 2 presents information about models, metrics and concepts related to MT and translation quality assessment. Section 3 contains a description of the experimental methodology, the prepared data set, and the results of the translation experiment. Section 4 reports the results of the experiment on fine-tuning models. Section 5 summarizes the results of the comparison with a discussion and concludes on the applicability of the considered models.

2 RELATED WORKS

This section provides an overview of the state-of-the-art research in the area of MT and related topics such as translation quality metrics and machine translation model fine-tuning,

2.1 Automatic Translations

The use of machine learning for automatic translation of texts is an actively developed area of research showing significant progress. Early statistical models based on analysis of word usage frequency show a result that poorly reflects the meaning of the original phrase. However, the development of artificial neural networks has significantly improved the quality of translation. Over the past years, the architecture of RNN networks has been used most often for MT (Mahata, Das, and Bandyopadhyay 2019; Vathsala and Holi 2020; Wang, Chen, and Xing 2019). However, such networks do not do a good job of keeping track of the context, and the longer the text, the worse they do, even when using LSTM (Vathsala and Holi 2020).

The quality of translation has been significantly improved as a result of the development of the Transformers architecture (Devlin et al. 2019; Wolf et al. 2020). In this architecture, the main focus is not on dependency tracking, but on the so-called attention mechanism in dependency calculation. Like RNN, the Transformers architecture is designed to transform one sequence into another, however, it does this not sequentially over the elements of the sequence, but over all the sequences at once.

Currently, most MT models are based on the Transformers architecture. For this work, the following models are considered: M2M100 (Fan et al. 2021), mBART (Liu et al. 2020), and OPUS-MT (Helsinki NLP) (Tiedemann and Thottingal 2020).

M2M100 is a multilingual encoder-decoder (seq-to-seq) model trained for Many-to-Many multilingual translation (Fan et al. 2021). The model is provided in two types: with 1.2 billion parameters and 418 million parameters. Both support 100 languages in total and allow for automatic translation between any pairs of languages.

Helsinki NLP is based on the MarianMT project that is essentially an add-on over the Marian C++ library for fast learning and translation. Using this library, a number of bilingual models have been trained by Helsinki University as well as two large models for translation from several languages into English and from English into several languages. List of supported languages contains 277 entries, excluding different alphabet for the same languages.

mBART is a sequence-to-sequence denoising auto-encoder pre-trained on large-scale monolingual corpora in many languages using the BART objective. mBART is one of the first methods for pre-training a complete sequence-to-sequence model by denoising full texts in multiple languages, while previous approaches have focused only on the encoder, decoder, or reconstructing parts of the text.

There are two versions of mBART: (1) *mbart-large-cc25* (with support for 25 languages) and (2) *mbart-large-50* (mBART-50, with support for 50 languages). mBART-50 is created using the original *mbart-large-cc25* checkpoint by extending its embedding layers with randomly initialized vectors for an extra set of 25 language tokens and then pre-trained on 50 languages.

2.2 Metrics

Since MT is primarily aimed at translating massive volumes of texts, manual evaluation of their quality is time-consuming and basically impractical (Tan et al. 2020). As a result, number of metrics have been developed for this purpose.

One of the most popular metrics is the BLEU score (Papineni et al. 2002) based on the comparison of n-grams of the one or several ground truth translations and the candidate (evaluated) translation. It is a universal language-independent metric.

Another popular metric is METEOR (Denkowski and Lavie 2014). It is also based on n-gram comparison, however unlike BLEU it takes into account language-specific information such as synonyms, paraphrasing, and stemmed forms of words. As a result, though in general the METEOR metric correlates with the BLEU metric, it is less strict and often closer to the human judgement. However, it is language-dependent and requires

additional language resources for translation evaluation.

2.3 Fine-tuning of Transformer Models

Fine-tuning of multilingual transformer models is a common practice. It is usually done to improve the models' performance in a given domain or for a given language. However, unlike, for example, fine-tuning of convolutional networks when most of the layers remain unchanged, fine-tuning of transformer networks usually assumes model training without freezing any layers, e.g. (Chen et al. 2021; Mishra, Prasad, and Mishra 2020).

Though normally, fine-tuning is done using relatively large training datasets (thousands of samples), in this particular research we will consider fine-tuning on small datasets. The reason for that is the absence of such a dataset or lack of time for its creation, when one needs to fine-tune the model only for several specific terms. The results of this study are presented in sec. 4.

3 ANALYSIS METHODOLOGY

3.1 Dataset Preparation

The dataset used contains 210,910 domain-specific texts in 32 languages, including English. When clearing the dataset, the following rules were used:

1. blank lines and extra spaces were removed;
2. references to objects attached to the source text were removed;
3. e-mails, records marked CONFIDENTIAL, and hyperlinks were removed;
4. all numbers were replaced by 1
5. all lines shorter than 20 characters were deleted.

After cleaning, the dataset contains 147866 texts in 32 languages.

3.2 Automatic Translation

To prepare a dataset, translate texts and calculate metrics, the system with Intel(R) Core(TM) i9-10900X CPU @ 3.70GHz and 64 Gb DDR4 was used. All services were launched in a virtual environment based on Docker with no restrictions on the amount of resources used.

The automatic translation process involves four models (M2M100-1.2B, M2M100-418M, mBART-50 and HelsinkiNLP mul2en) sequentially translating the data set from the original language into English. The original language is determined automatically by

the process described in section 3.3.1. The Huggingface Transformers library is used to run the models. The result of the translation is saved to the data set. After the processing is completed, the received translations are evaluated using the BLEU and METEOR metrics and visualization is carried out for their manual analysis.

Additionally, an experiment was carried out with the division of each text into paragraphs and independent translation of paragraphs with a preliminary definition of the language. This is due to the specifics of the problem area. Each text can contain parts in different languages, and automatic language detection identifies the most frequently encountered language, which can significantly worsen the translation result. To circumvent this situation, a solution was proposed with the division of the text into paragraphs.

3.3 Translation with MS Azure Cloud

Due to large dataset on 32 languages it is highly difficult to create reference translation manually. The Translation service, which is part of the Cognitive Services of the Microsoft Azure platform, was chosen as a benchmark metric for translating texts. It should be noted that Azure translation is also not ideal and a better choice would be to compare with text translated by professional translators. Therefore this experiment should be interpreted solely as a comparison of the translation quality of the selected models between each other relating to the translation of MS Azure Translator.

A free Tier was used, which provides the ability to translate up to 2,000,000 characters per month. At the same time, there is an additional limitation on the speed of access to the server, which should not exceed 33300 characters per minute.

Since there is a limit of 2 million characters for using the translation service, the limit of 70 texts was set for each language. This gave a total of 1,658 examples with a total of 1,819,898 characters. The distribution of the lengths of all texts is shown in Figure 1. Hereinafter, when displaying statistical results, a box plot will be used.

In order to meet the limit on the number of characters per minute, a delay was set in the cycle in which the translation takes place, depending on the length of the text that was submitted for translation. The delay duration is calculated as the nearest integer when the text length is divided by 550 (maximum number of characters per second).

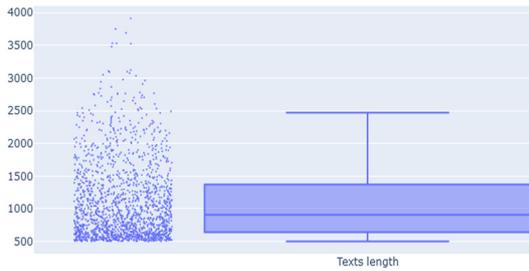


Figure 1: Distribution of texts in dataset by length.

3.3.1 Language Detection

Since the task is to translate texts into English, then from the remaining texts it is necessary to select those that are presented in languages other than English. For this, four utilities were used to determine the text language: langdetect, langid, fasttext, and cld3. Each text was processed by all four utilities and the results were placed in the appropriate columns. Of the entire dataset, a language definition conflict (a situation when at least one of the utilities gives a different result from the others) was found for 24393 examples. For example, for some of the texts it could be a situation, when the used utilities provide set of languages like ('en', 'de', 'fr', 'nl'). In this case it is impossible to detect language, since not all models provide their confidence levels. If at least two models identify the same languages (i.e., set ('en', 'en', 'de', 'fr')) then this language is chosen as the language text ('en' in this example). For the subsequent work, the rows of the dataset were selected, for which all four utilities determined the same language. Since translation into English requires text in a language other than English, additional filtering was carried out. This was done using a filter over the data frame to select rows that do not contain English language detected by the consensus of all four models.

According to the results of the mask selection, 90219 texts remained. To improve the quality of the translation, it was decided to select texts over 500 characters long. This is justified by the need to choose the language for the text and to study the influence of

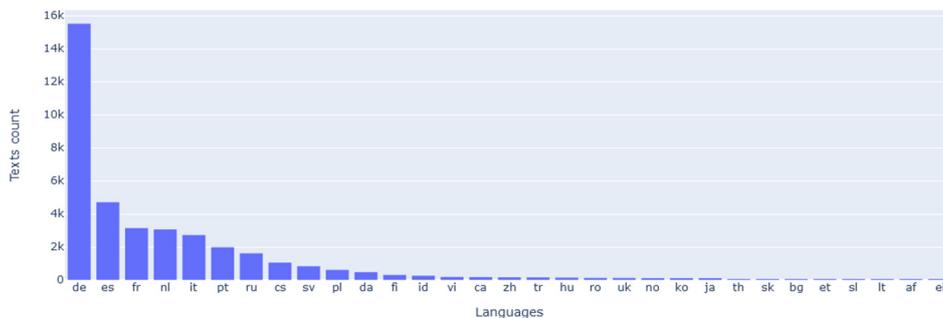


Figure 2: Dataset distribution by language after filtering by text length.

the context on the accuracy of the translation. After filtering texts by length, the distribution turned out as follows. The distribution of dataset texts is shown in Figure 2.

3.3.2 Metric Evaluation

The general BLEU metric is shown in Figure 3 (left) using the box plot. The figure shows that the translation by the model M2M100-1.2B (BLEU - 0.51) is the closest to the reference text. HelsinkiNLP has the lowest rating (BLEU - 0.27).

For the METEOR metric, the relative results were the same. Model M2M100-1.2B is the leader with METEOR = 0.74 and HelsinkiNLP has the lowest score with METEOR = 0.56. The absolute results of the METEOR metrics are higher due to the use of synonyms when comparing texts. The distribution of ratings is shown on Figure 3 (right).

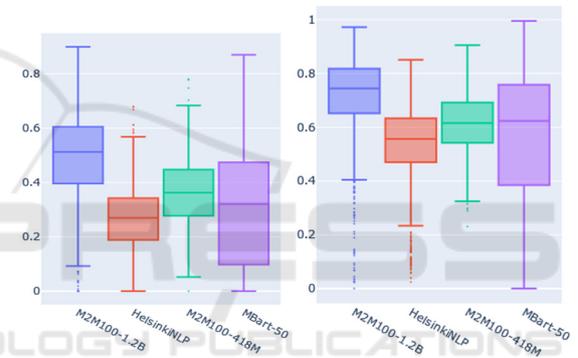


Figure 3: BLEU (left) and METEOR (right) metrics comparison by model.

For translation by paragraphs, the BLEU and METEOR metrics were also calculated (Fig 4). The results were noticeably lower than when translating whole texts. Presumably this is due to several reasons. First, for shorter texts, the accuracy of determining the language may be lower than for large texts. Hence, the translation quality may be lower, since many models need to specify the source language.

This assumption is justified by the fact that in some cases (2651 out of 23928) the language defined by the models did not match the language defined by the MS Azure service. Another possible reason is that when translating a multilingual text, part of it may remain untranslated, both when translating with MS Azure and when translating with the models under study. Since the text is not translated, it is simply transferred to the translation result, which is why it is possible that the same parts of the text appear, which is perceived by the metrics as a correct translation and the estimate is overestimated.

In the case of paragraph translation, a smaller division of the text occurs that leads to language detection for each part, which will subsequently be translated. Hence, the score may be lower.

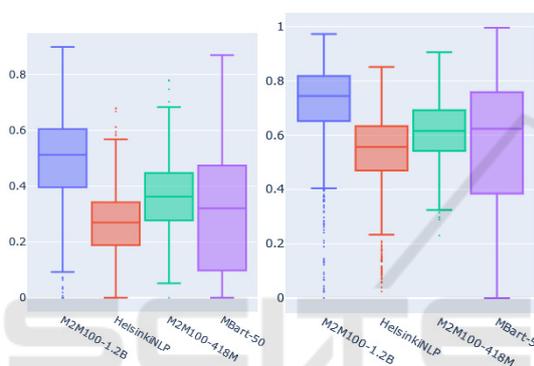


Figure 4: BLEU (left) and METEOR (right) metrics comparison by model for paragraphs translation.

Table 1 shows the estimates of the models in terms of execution time and average metrics of translation quality.

Table 1: Comparison of model indicators.

Model	Translation time	Translation time by paragraph	BLEU median	METEOR Median	BLEU median by paragraph	METEOR Median by paragraph
MarianMT / HelsinkiNLP	4:09	4:55	0,27	0,56	0,22	0,65
M2M100 1.2B (large)	22:50	23:58	0,51	0,74	0,39	0,73
M2M100 418M (small)	8:47	10:56	0,36	0,61	0,35	0,71
mBART50	10:40	11:29	0,30	0,60	0,00	0,57

4 FINE-TUNING

This section reports results of experiments aimed at studying how translation transformer models can be fine-tuned for correct translation of specific terms

without preparing large textual datasets consisting of multiple (thousands) samples but only using samples with the corresponding terms.

For experimenting we have selected specific terms, which are usually are not translated correctly by the MS Azure service normally used. The terms are presented in Table 2. The corresponding validation set have been collected including both sentences, which contain the terms being fine-tuned, and sentences with similar wordings where the words are not part of the specific term and should be translated as in usual spoken language. The goal of this is not only evaluate how translation models are fine-tuned, but also to make sure that the normal translation is not broken by the fine-tuning procedure.

The fine-tuning is done as a regular training process, which is terminated when the BLEU metric stops to improve on the validation set.

Table 2: Training set (s – singular form, p – plural form).

Form	Source text (de)	Source text (ru)	Target (the ground truth) text (en)
s	die Sensornut	паз для датчиков	sensor slot
s	die Endlagendämpfung	фиксированное демпфирование	end-position cushioning
s	Normzylinder	стандартизированный цилиндр	standard cylinder
s	Ventilinsel	пневмоостров	valve manifold
s	der Antrieb	привод	actuator
p	Sensornuten	пазы для датчиков	sensor slots
p	der Normzylinder	стандартизированные цилиндры	standard cylinders
p	Ventilinseln	пневмоострова	valve manifolds
p	Antriebe	приводы	actuators

4.1 Fine-tuning on a Reduced Training Dataset

This experiment is aimed at studying possibilities to fine-tune models based on the usage of minimum data. We try to use the reduced training set consisting only of Russian translations in singular (“sing” in the table) form. However, the fine-tuned model is then applied to both Russian-to-English and German-to-English translations to see if fine-tuning for one language affects translation for another language. The results are shown in Figures 5-8.

It was concluded that M2M100 models cannot be fine-tuned this way. They require fine-tuning based on large textual datasets consisting of multiple

sentences. One can see that the quality of translation does not increase after such fine-tuning, and sometimes even decreases. These models tend to shorten the output to the terms used for training: e.g., “Ein Normzylinder hat zwei Sensornuten.” is translated as “standard cylinder” after several epochs.

Helsinki NLP and mBART-large models can generally be successfully fine-tuned. The metrics increase significantly.

Helsinki NLP and mBART-large models are able to transfer the terminology used for fine-tuning between languages. For example, after training only on Russian singular terms, the German translation changes the following way:

Text to be translated: “Anfrage der technischen Dokumentation zu MPA Ventilinsel.”

Original (before fine-tuning) translation: “Application for technical documentation to MPA Ventilinsel.”

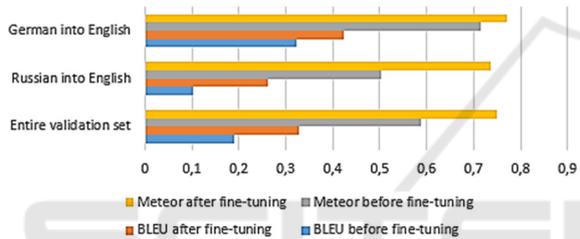


Figure 5: Results of Helsinki-NLP fine-tuning on a reduced dataset.

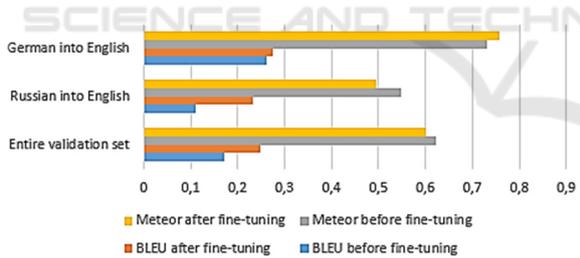


Figure 6: Results of M2M100 418M fine-tuning on a reduced dataset.

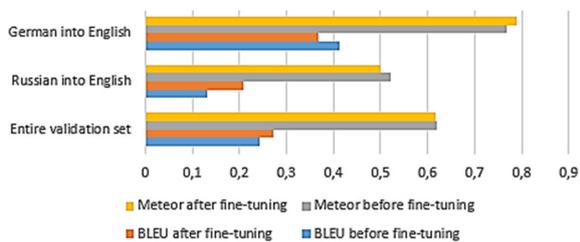


Figure 7: Results of M2M100 1.2B fine-tuning on a reduced dataset.

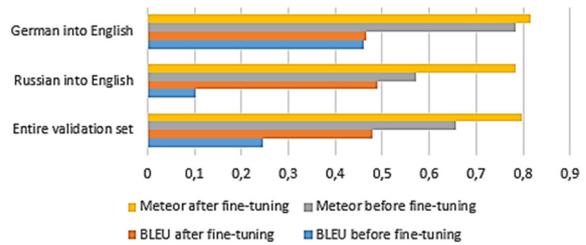


Figure 8: Results of mBART-large fine-tuning on a reduced dataset.

The ground truth translation: “Request for technical documentation about MPA valve manifold.”

Translation after fine-tuning: “request of technical documentation to MPA valve manifold.”

4.2 Fine-tuning on the Complete Training Dataset

This experiment is aimed at the analysis of what translation result can be achieved on the whole training dataset provided. The results are shown in Figures 9-12.

It was concluded that M2M100 models still cannot be fine-tuned this way. They require fine-tuning based on large textual datasets consisting of multiple sentences. One can see that the quality of translation does not increase after such fine-tuning, and sometimes even decreases. These models still tend to shorten the output to the terms used for training (e.g., “Ein Normzylinder hat zwei Sensornuten.” is translated as “standard cylinder” after several epochs).

Helsinki NLP and mBART-large models can generally be successfully fine-tuned. The metrics increase significantly (and even better than in the first experiment).

5 CONCLUSION & DISCUSSION

Of the models reviewed, the M2M100 1.2B model is the undisputed leader in translation quality. Regardless of the translation method, this particular model has the highest rates in all metrics. The only drawback is the large size and long translation time (almost a day on the presented dataset). The rest of the models, on average, show fairly similar results.

The mBART-50 model showed very low results in translation, associated with duplication of texts when translating in a loop and with the replacement of the target language when translating by paragraphs. It was not possible to establish the cause of this behavior, and

therefore it is necessary to conduct a separate series of experiments to determine the causes.

The only model among the reviewed ones that can detect language by its own is Helsinki NLP. The model is trained to automatically recognize tokens in all supported languages and recover text from tokens in the target language for Many-To-One languages model and provide only target language for One-To-Many or Many-To-Many model. All other models necessarily require specifying the language for the tokenizer and the target language for translation.

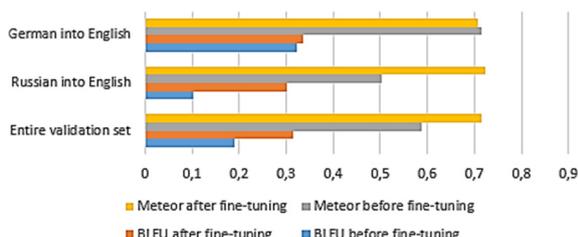


Figure 9: Results of Helsinki-NLP fine-tuning on the complete dataset.

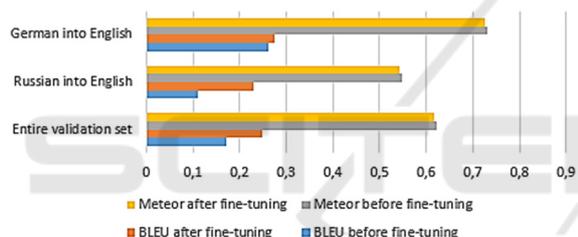


Figure 10: Results of M2M100 418M fine-tuning on the complete dataset.

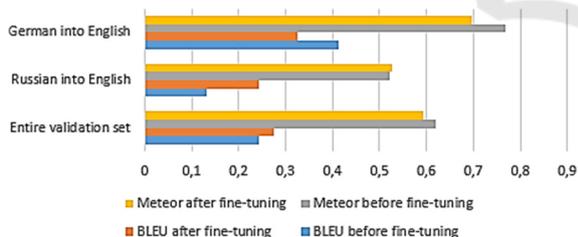


Figure 11: Results of M2M100 1.2B fine-tuning on the complete dataset.

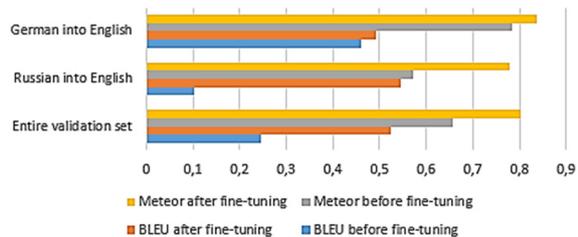


Figure 12: Results of mBART-large fine-tuning on the complete dataset.

There are several factors to consider when deciding which model to use for automatic translation. If translation time is critical, then the MarianMT / HelsinkiNLP model provides the best result with an acceptable translation quality. Moreover, it provided the growth of the METEOR metric when translated by paragraphs. If you need to ensure the maximum quality of translation, then here the clear choice is M2M100 1.2B (Large). At the same time, the simplified model M2M100 418M (small) is a reasonable compromise, providing satisfactory translation quality and a fairly short time.

If the text contains words or sentences in a language other than the language of the entire text, there may be problems associated with tokenization and translation. Such cases can be handled differently depending on the model. Unknown tokens may be excluded from the result or the model may give an incorrect translation (M2M100-1.2B & M2M100-M418), the model may try to tokenize and translate the text (HelsinkiNLP & mBART-50), sometimes with an unpredictable result.

The algorithm for dividing the text into paragraphs can only be beneficial if the paragraphs contain more than 10 words. In this case, the number of errors associated with incorrect language definition is reduced due to the larger number of examples. This is due to the borrowing of words between languages and related language groups (eg, Indo-European language family), between which it is sometimes difficult to distinguish (eg Spanish-Italian). Thus, for short paragraphs, it is recommended not to split them into separate paragraphs, but to define the text for the entire text and then translate it. For long paragraphs (more than 10 words each), the text can be divided into separate paragraphs and translated independently.

Regarding the fine-tuning for specific terminology on small datasets, it was found that M2M100 models cannot be fine-tuned using only short terms/phrases and they require large datasets consisting of multiple texts (thousands of training pairs) with the required terminology. At the same time both Helsinki NLP and mBART-large models can be fine-tuned successfully. They are also able to transfer the terminology used for fine-tuning between languages. As a result, one can fine-tune models for new terms only for few languages and the other languages will likely be translated correctly in accordance with the new terminology. However, increasing the number of training languages increases the quality of translation after fine-tuning.

The validation set is an important factor of successful fine-tuning. It should consist of both

sentences with the terms being fine-tuned and without them. This will make it possible to better evaluate when the fine-tuning should be stopped (the models have learnt the new terms and other terminology is not damaged). The larger validation set will provide better evaluation of the fine-tuning results.

It was also observed that unlike the METEOR metric, the BLEU metric is very strict and often is equal 0 even if the translation is correct. On the contrary, the Meteor metric may not take into account the correctness of particular terms since it applies synonyms to the evaluation. As a result, it is recommended to use the BLEU metric on a reasonably large validation set for evaluation of early stopping point during fine-tuning, and the Meteor metric for the final evaluation of the translation quality.

ACKNOWLEDGEMENTS

The paper is due to the collaboration between SPC RAS and Festo SE & Co. KG. The methodology and experiment setup (sec. 3) are partially due to the State Research, project number FFZF-2022-0005.

REFERENCES

- Chen, Ben et al. 2021. "Transformer-Based Language Model Fine-Tuning Methods for COVID-19 Fake News Detection." *Communications in Computer and Information Science* 1402: 83–92.
- Cioffi, Raffaele et al. 2020. "Artificial Intelligence and Machine Learning Applications in Smart Production: Progress, Trends, and Directions." *Sustainability* 12(2): 492.
- Denkowski, Michael, and Alon Lavie. 2014. "Meteor Universal: Language Specific Translation Evaluation for Any Target Language." In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, Stroudsburg, PA, USA: Association for Computational Linguistics, 376–80.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." In *Proceedings of NAACL-HLT 2019*, 4171–86.
- Fan, Angela et al. 2021. "Beyond English-Centric Multilingual Machine Translation." *Journal of Machine Learning Research* 22: 1–48..
- Google. 2022. "Google Translate."
- Jooste, Wandri, Rejwanul Haque, and Andy Way. 2021. "Philipp Koehn: Neural Machine Translation." *Machine Translation* 35(2): 289–99.
- Liu, Yinhan et al. 2020. "Multilingual Denoising Pre-Training for Neural Machine Translation." *Transactions of the Association for Computational Linguistics* 8: 726–42..
- Mahata, Sainik Kumar, Dipankar Das, and Sivaji Bandyopadhyay. 2019. "MTIL2017: Machine Translation Using Recurrent Neural Network on Statistical Machine Translation." *Journal of Intelligent Systems* 28(3): 447–53.
- Microsoft. 2022. "Microsoft Bing Translator."
- Mishra, Sudhanshu, Shivangi Prasad, and Shubhanshu Mishra. 2020. "Multilingual Joint Fine-Tuning of Transformer Models for Identifying Trolling, Aggression and Cyberbullying at TRAC 2020." In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying at Language Resources and Evaluation Conference (LREC 2020)*, European Language Resources Association (ELRA), 120–25.
- Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. "BLEU: A Method for Automatic Evaluation of Machine Translation." In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, 311–18.
- Stahlberg, Felix. 2020. "Neural Machine Translation: A Review." *Journal of Artificial Intelligence Research* 69: 343–418.
- Tan, Zhixing et al. 2020. "Neural Machine Translation: A Review of Methods, Resources, and Tools." *AI Open* 1: 5–21.
- Tiedemann, Jörg, and Santhosh Thottingal. 2020. "OPUS-MT: Building Open Translation Services for the World." In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, 479–80.
- Usuga Cadavid, Juan Pablo et al. 2020. "Machine Learning Applied in Production Planning and Control: A State-of-the-Art in the Era of Industry 4.0." *Journal of Intelligent Manufacturing* 31(6): 1531–58.
- Vathsala, M. K., and Ganga Holi. 2020. "RNN Based Machine Translation and Transliteration for Twitter Data." *International Journal of Speech Technology* 23(3): 499–504.
- Wang, Xu, Chunyang Chen, and Zhenchang Xing. 2019. "Domain-Specific Machine Translation with Recurrent Neural Network for Software Localization." *Empirical Software Engineering* 24(6): 3514–45.
- Wolf, Thomas et al. 2020. "HuggingFace's Transformers: State-of-the-Art Natural Language Processing." In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Stroudsburg, PA, USA, ACL, 38–45.
- Zhang, JiaJun, and ChengQing Zong. 2020. "Neural Machine Translation: Challenges, Progress and Future." *Science China Technological Sciences* 63(10): 2028–50.