ADAM MICKIEWICZ UNIVERSITY, POZNAŃ
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Artur Nowakowski

Quality Optimization Methods in Neural Machine Translation Systems

Doctoral thesis

Supervisor:
prof. dr hab. Krzysztof Jassem

Auxiliary supervisor:
dr Maciej Lison

Discipline:
Computer and Information Sciences

Poznań, 2023

UNIWERSYTET IM. ADAMA MICKIEWICZA W POZNANIU
WYDZIAŁ MATEMATYKI I INFORMATYKI

Artur Nowakowski

Metody optymalizacji jakości w neuronowych systemach tłumaczenia maszynowego

Rozprawa doktorska

Promotor:
prof. dr hab. Krzysztof Jassem

Promotor pomocniczy:
dr Maciej Lison

Dyscyplina:
Informatyka

Poznań, 2023

Acknowledgments

I would like to express my sincere gratitude to my supervisor prof. Krzysztof Jassem, who has been a constant source of support and guidance throughout my PhD journey. Your constructive feedback has been invaluable in helping me complete this thesis. I am deeply grateful for the time and effort you have put into reading and reviewing my work, and for your invaluable advice and insights.

I would also like to thank my auxiliary supervisor dr Maciej Lison, for their guidance and support during the course of my PhD. Your insights and expertise have been greatly appreciated and have significantly contributed to this thesis.

I would like to extend my heartfelt thanks to my family and friends for their love, support and encouragement. Their belief in me has been a source of inspiration and strength during this time.

Finally, I would like to express my deepest gratitude to my fiancée Gabriela, for her unwavering love and support throughout this journey. Your encouragement and understanding have been a source of comfort and motivation, and I am truly grateful to have you in my life.

Abstract

The thesis presents new quality optimization methods in neural machine translation systems. It is based on seven scientific papers presented at international conferences.

Chapter 1 introduces the research problem, motivation, structure and scope of the thesis. It provides an overview of the included papers, together with details on authors, venues, presentation type, and the contribution of the thesis author. The chapter also contains a short description of each paper included in the thesis.

Chapters 2 to 5 present research papers on quality optimization methods in neural machine translation systems. They also include descriptions of methods applied in the solutions of shared tasks held at scientific conferences. Chapter 2 introduces a new method for integrating hand-crafted lexicons in machine translation involving morphologically rich languages. Chapter 3 reports on the methods used in the solutions of shared tasks organized as part of the PolEval 2021 workshop on translation quality estimation and evaluation. Chapters 4 and 5 describe the methods used in the solutions of shared tasks organized as part of the WMT 2021 and WMT 2022 conferences.

Chapters 6 to 8 present development papers that describe real-world neural machine translation systems developed as part of participation in the Industrial PhD program. Chapter 6 reports on the machine translation system created for the Polish Border Guard within the R&D project "Advanced Internet analysis supporting the detection of criminal groups". Chapter 7 discusses the challenges encountered in implementing and deploying a machine translation system for the EY corporation. Chapter 8 describes POLENG MT, an adaptive machine translation platform that can be used as a cloud-based web application or as an on-site solution.

The appendices include a certificate from the WMT 2022 conference organizers and declarations of contribution from the co-authors of each paper.

Streszczenie

Rozprawa doktorska przedstawia nowe metody optymalizacji jakości w neuronowych systemach tłumaczenia maszynowego. Praca składa się z siedmiu artykułów naukowych zaprezentowanych podczas konferencji o zasięgu międzynarodowym.

Rozdział 1 opisuje problem badawczy, motywację, strukturę i zakres rozprawy. Zawiera przegląd oraz krótki opis załączonych artykułów, w tym informacje o autorach, miejscu i typie prezentacji, a także wkładzie autora rozprawy.

Rozdziały od 2 do 5 prezentują prace badawcze dotyczące metod optymalizacji jakości w neuronowych systemach tłumaczenia maszynowego. Zawierają również opisy metod zastosowanych w rozwiązaniach konkursów organizowanych w ramach konferencji. W rozdziale 2 przedstawiono nową metodę integracji leksykonów w tłumaczeniu maszynowym, mającą zastosowanie dla języków fleksyjnych. Rozdział 3 opisuje metody oceny jakości tłumaczenia zastosowane w rozwiązaniach konkursów organizowanych w ramach warsztatu PolEval 2021. Rozdziały 4 i 5 opisują metody zastosowane w rozwiązaniach konkursów organizowanych w ramach międzynarodowych konferencji WMT 2021 i WMT 2022.

W rozdziałach 6 do 8 przedstawiono artykuły opisujące prace rozwojowe. W ramach artykułów opisano neuronowe systemy tłumaczenia maszynowego opracowane w trakcie doktoratu wdrożeniowego. Rozdział 6 opisuje system tłumaczenia maszynowego stworzony dla Straży Granicznej w ramach projektu badawczo-rozwojowego „Zaawansowana analiza Internetu wspomagająca wykrywanie grup przestępczych". W rozdziale 7 omówiono wyzwania, jakie napotkano w implementacji i wdrażaniu systemu tłumaczenia maszynowego dla korporacji EY. Rozdział 8 opisuje system o nazwie POLENG MT – adaptacyjną platformę tłumaczenia maszynowego, która może być wykorzystywana jako aplikacja internetowa w chmurze lub jako rozwiązanie instalowane w infrastrukturze klienta.

W załącznikach zamieszczono certyfikat otrzymany od organizatorów konferencji WMT 2022 oraz deklaracje o wkładzie współautorów każdego artykułu.

Contents

1 Introduction

Neural Machine Translation (NMT) is a type of machine translation that utilizes neural networks to translate text from one language to another. In recent years, there have been significant advancements in NMT, which have led to improved translation quality and efficiency.

One of the major advancements in NMT is the use of the attention mechanism and the Transformer (Vaswani et al., 2017) neural network architecture, which allows the model to focus on specific parts of the input sentence during translation. This helps the model to better understand the meaning of the sentence and produce more accurate translations. Additionally, the use of pre-trained neural language models, which are trained on a large corpus of text data before being fine-tuned for specific tasks, has also led to improvements in NMT and its evaluation processes (Freitag et al., 2022; Liu et al., 2020).

Despite the advancements, there are still shortcomings that need to be addressed in specific translation scenarios. These include, for example, incorporating human knowledge into NMT models, taking into account the context of the entire document, and ensuring accurate translation of named entities.

Currently, NMT models are considered as black-box solutions that are trained on large datasets consisting sentence pairs in the source and target languages. Therefore, it proves difficult to integrate human knowledge, such as hand-crafted lexicons, into the machine translation process. The ability to do so is crucial, especially in the case of domain-specific translations, such as biomedical, legal, criminal, or e-commerce, where it is required that terminology should be translated in a predefined way.

The focus in NMT is shifting from standard sentence-level models to document-level models. Document-level machine translation allows for the translation of entire documents, such as books, articles, and reports, rather than just individual sentences or phrases. It enables a more complete understanding of the content being translated, as the context provided by the surrounding text can help to disambiguate meaning, improve fluency and coherence of the translated text, which is especially important for longer documents. This is a more complex task than translating individual sentences in isolation, as the model must take into account the relationships and dependencies between sentences and the overall meaning of the document. Furthermore, document-level machine translation models often require large amounts of document-level training data and computational resources, making them more difficult and expensive to develop and maintain.

There are various methods for enhancing the quality of machine translation, which can be determined by factors such as the intended application, the availability of training data, and the unique characteristics of the language in question.

The goal of this thesis is to propose new methods for quality optimization in NMT, as well as to demonstrate their application in real-world machine translation systems.

1.1 Motivation, Structure and Scope of the Thesis

Motivation

The thesis is the outcome of the Industrial PhD (pol. *doktorat wdrożeniowy*) program, initiated by the Polish government in 2017. An Industrial PhD is a type of PhD that is focused on applied research and involves collaboration between the student, a university, and an industry partner. The program is designed to provide students with the opportunity to conduct research that is directly relevant to the needs of industry, and to develop the skills and knowledge needed to work in industry or academia.

This thesis is the result of a collaboration with the Poleng company and Adam Mickiewicz University. Since its establishment in 2004, Poleng has been involved in the development of machine translation systems, including the pioneering Translatica rule-based machine translation system. By collaborating with Adam Mickiewicz University in the Industrial PhD program, the company aimed to ensure the highest possible quality of solutions provided by machine translation systems utilizing state-of-the-art methods. The results obtained during the PhD have been deployed in Poleng, as shown in Chapter 7 (nEYron: Implementation and Deployment of an MT System for a Large Audit & Consulting Corporation) and Chapter 8 (POLENG MT: An Adaptive MT Platform).

Structure and Scope

Due to the nature of the PhD, the thesis is divided into two sections: *research papers* and *development papers*.

The *research papers* section comprises four papers that are strictly related to research and participation in shared tasks. The papers included in this section describe new approaches to quality optimization methods in NMT. Each paper presents original ideas for optimizing NMT quality in various scenarios, such as translation between low-resource language pairs, the incorporation of domain-specific terminology, the use of translation quality estimation models to improve NMT, document-level translation, and proper translation of named entities.

The *development papers* section concentrates on the application of research ideas and experiments in real-world MT systems. It comprises three papers, each of which describes a different real-life use case and translation scenario. Two of the papers are closely associated with the work at the Poleng company, and one is related to the MT system developed as a result of participation in an R&D project.

A full list of the papers included in the thesis can be found in Table 1.1. The table also includes information on the venue where the paper was published and the awarded MEiN points. A brief overview, including the motivation behind each paper, is provided in Section 1.2.

List of Papers

Table 1.1: List of papers included in the thesis

Title	Authors	Venue	MEiN Points
Neural Machine Translation with Inflected Lexicon	A. Nowakowski, K. Jassem	MT Summit 2021	70
Approaching English-Polish Machine Translation Quality Assessment with Neural-based Methods	A. Nowakowski	PolEval 2021	20
Adam Mickiewicz University's English-Hausa Submissions to the WMT 2021 News Translation Task	A. Nowakowski, T. Dwojak	WMT 2021 (EMNLP 2021)	140
Adam Mickiewicz University at WMT 2022: NER-Assisted and Quality-Aware Neural Machine Translation	A. Nowakowski*, G. Pałka*, K. Guttmann†, M. Pokrywka†	WMT 2022 (EMNLP 2022)	140
A Neural Translator Designed to Protect the Eastern Border of the European Union	A. Nowakowski, K. Jassem	MT Summit 2021	70
nEYron: Implementation and Deployment of an MT System for a Large Audit & Consulting Corporation	A. Nowakowski, K. Jassem, M. Lison, R. Jaworski, T. Dwojak, K. Wiater, O. Posesor	EAMT 2022	70
POLENG MT: An Adaptive MT Platform	A. Nowakowski, K. Jassem, M. Lison, K. Guttmann, M. Pokrywka	EAMT 2022	70

1.2 Papers Overview

The papers included in this thesis were prepared and published between the years 2021-2022. Seven papers were presented as oral presentations or posters at international venues related to MT and have been included in the proceedings volumes of these venues.

As the main author of all the papers included in the thesis, I was responsible for the conceptualization and methodology of the research work in each of them. Specific contributions to each paper are listed in the overview and in the declarations of contribution (see Appendix B).

Research Papers

1. **Neural Machine Translation with Inflected Lexicon**

Authors:
Artur Nowakowski, Krzysztof Jassem

Venue:
Machine Translation Summit 2021 (Virtual)

Presentation type, date (presenters):
Oral presentation, 19.08.2021 (Artur Nowakowski)

Published paper URL (accessed 16.01.2023):
https://aclanthology.org/2021.mtsummit-research.23.pdf

Author contribution:
Conceptualization and methodology of the research work, algorithm for inflected lexical constraints incorporation, implementation and experimental setup, conduct of the experiments, human and automatic evaluation analysis, writing of the paper.

The integration of human knowledge with a hand-crafted lexicon is a common need in NMT. However, it has proven to be a challenge, particularly for morphologically rich languages like Polish or Ukrainian. In these cases, it is not only crucial to include the base form of words or phrases in the translated sentence, but also to inflect them correctly while taking into account the surrounding context.

The paper describes experiments in NMT that involve lexical constraints for the English-Polish translation direction. A method based on a constrained decoding algorithm was introduced to address inflected forms of lexical entries without altering the training data or the model architecture. The proposed method was evaluated in both general and domain-specific scenarios and compared to the baseline translation in terms of quality and efficiency. New evaluation metrics were proposed to evaluate the method's ability to handle constraints, taking into account factors such as the presence, placement, duplication, and inflectional correctness of lexical terms in the output sentence.

We demonstrated that our method for enforcing the incorporation of terminology from a lexicon improves translation quality in domain-specific scenarios, but it diminishes it in general scenarios where the translation of specific terms is uncertain. Our method placed and inflected terminology correctly in almost all of cases, but at the expense of translation speed, when compared to the baseline solution.

2. **Approaching English-Polish Machine Translation Quality Assessment with Neural-based Methods**

Authors:
Artur Nowakowski

Venue:
PolEval 2021 (Virtual)

Presentation type, date (presenters):
Publication without the presentation

Published paper URL (accessed 16.01.2023):
http://poleval.pl/files/poleval2021.pdf

Author contribution:
Conceptualization and methodology of the research work, conduct of the experiments with Polish-English neural machine translation quality assessment models, automatic evaluation analysis, writing of the paper.

Machine Translation Quality Assessment is the task of assessing the quality of machine-translated text. This can be accomplished through the use of a reference translation (machine translation evaluation) or without one (machine translation quality estimation).

Before the rise of large language models (LLMs), the evaluation of machine translation relied on lexical metrics that required human reference translations, such as BLEU (Papineni et al., 2002) or chrF (Popović, 2015). However, as the quality of machine translation has improved over the years, these metrics have not kept pace. The reliance on a single reference translation meant that machine translation systems that produced good translations which were different from the reference translations, were penalized by these metrics.

The current state-of-the-art machine translation metrics, such as COMET (Rei et al., 2020), are LLMs that have been fine-tuned on human judgments for the task of evaluating machine translation. These metrics are more robust and can capture the context of the sentence, meaning that different equally good translations are scored accordingly well. They can also be trained in different scenarios, so that they can work with or without human references (Wan et al., 2022).

Excluding human references in the model training process has shown that neural-based translation quality estimation models can estimate translation quality effectively even without access to a reference translation, and have a stronger correlation with human judgments than standard lexical metrics that rely on references (Freitag et al., 2022).

The use of decoding algorithms, such as beam search (Sutskever

et al., 2014) or *p*-nucleus sampling (Holtzman et al., 2020), allows an MT system to produce multiple hypotheses. The neural-based quality estimation models can then be used to select the best hypothesis from a set of multiple hypotheses, in a way that simulates a human judgment, as shown in Chapter 5 (Adam Mickiewicz University at WMT 2022: NER-Assisted and Quality-Aware Neural Machine Translation).

The paper presents experiments for the PolEval 2021 machine translation quality assessment shared task, including both subtasks: reference-based (*nonblind*) and reference-less (*blind*) quality assessment.

We describe experiments in which we fine-tuned LLMs using state-of-the-art neural network architectures for the English-Polish translation direction. Our results indicate that fine-tuning LLMs for a specific language pair can further increase the correlation (measured by Pearson's *r*) with human judgments. Additionally, we found that utilizing LLMs that have been pre-trained specifically for the Polish language, such as HerBERT (Mroczkowski et al., 2021), can give better results for language pairs involving Polish than using multilingual LLMs.

3. **Adam Mickiewicz University's English-Hausa Submissions to the WMT 2021 News Translation Task**

Authors:
Artur Nowakowski, Tomasz Dwojak

Venue:
Sixth Conference on Machine Translation (Virtual / Punta Cana, Dominican Republic)

Presentation type, date (presenters):
Poster presentation, 10.11.2021 (Artur Nowakowski, Tomasz Dwojak)

Published paper URL (accessed 16.01.2023):
https://aclanthology.org/2021.wmt-1.14.pdf

Author contribution:
Conceptualization and methodology of the research work, implementation of the data filtering steps, conduct of the experiments with iterative back-translation, PB-SMT and model ensembling, writing of the paper.

There are a number of reasons why machine translation can be challenging for low-resource languages. One major challenge is the lack of training data. Most machine translation models rely on large amounts of parallel text data in order to learn how to translate between languages. However, for low-resource languages, there may not be a sufficient amount of parallel text data available, making it difficult for the model to learn how to translate accurately.

Moreover, low-resource languages may have a smaller number of speakers, which can make it difficult to gather enough native speakers to evaluate and improve the machine translation systems.

The paper presents the submissions of Adam Mickiewicz University to the WMT 2021 News Translation Task in the English-Hausa and Hausa-English translation directions, which is a low-resource translation scenario between distant languages. We experimented with multiple methods to optimize the quality of the translation for low-resource languages. These methods include transfer learning from a high-resource language pair, iterative back-translation to systematically expand the training data, and thorough data filtering. It is particularly important to thoroughly filter the data when working with low-resource languages, as the model is more likely to produce inaccurate translations if most of the training dataset is noisy.

We experimented with both NMT and PB-SMT (Phrase-based Statistical Machine Translation) and found that the translation quality was comparable when the training dataset was not synthetically expanded. Our approach significantly improved the translation quality compared to the baseline solution. The gains were particularly noticeable in the English-Hausa translation direction, as our system ranked fourth among the constrained submissions (Akhbardeh et al., 2021).

4. **Adam Mickiewicz University at WMT 2022: NER-Assisted and Quality-Aware Neural Machine Translation**

Authors:
Artur Nowakowski[*], Gabriela Pałka[*], Kamil Guttmann[†], Mikołaj Pokrywka[†]

* and † denote equal contribution groups

Venue:
Seventh Conference on Machine Translation (Abu Dhabi, United Arab Emirates)

Presentation type, date (presenters):
Poster presentation, 07.12.2022 (Artur Nowakowski, Gabriela Pałka, Kamil Guttmann, Mikołaj Pokrywka)

Published paper URL (accessed 16.01.2023):
https://aclanthology.org/2022.wmt-1.26.pdf

Author contribution:
Conceptualization and methodology of the research work, idea behind the system as a whole, integration of the separate components into a single system, implementation of the data filtering process, conduct of the experiments with transfer learning, back-translation, quality-aware decoding and model ensembling, writing of the paper.

Creating an MT system for languages that use different alphabets (e.g. Latin and Cyrillic), can present a number of difficulties. For example, rare words and named entities, such as names, surnames, places, organizations, are often mistranslated between such languages. Named entities are often culturally specific and may not have direct equivalents in the target language. For example, a street name or a famous historical figure may not exist in the target language and would require special handling by the MT system. Another difficulty with named entities is that they often have proper noun capitalization, which can be lost when translating to languages with different capitalization rules. Furthermore, some named entities may be ambiguous, meaning they can refer to multiple entities. For example, "Róża" can be a proper name of a person or a common noun referring to a flower.

Finally, due to the complexity and variability of named entities, the model would need to be trained with a large and diverse set of named entity examples in order to generalize well to unseen examples. However, this can prove difficult in the case of low-resource languages, where training data is scarce. As such, knowledge about named entities needs to be incorporated into the model in alternative ways that are not dependent on the size of the dataset.

The paper describes Adam Mickiewicz University's submissions to the constrained track of the WMT 2022 General MT Task. We participated in the Ukrainian-Czech and Czech-Ukrainian translation directions, which is a translation scenario between low-resource languages using different alphabets. Our experiments include multiple novel quality optimization methods for NMT, such as transfer learning, back-translation, NER-assisted translation, document-level translation, quality-aware decoding, model ensembling, on-the-fly domain adaptation, advanced data filtering and post-processing.

We were able to successfully incorporate all of these methods, except for on-the-fly domain adaptation, into a single MT system. Each method contributed into the final result, some more significantly than others. Our greatest advancements in quality, as measured by automatic evaluation metrics, came from NER-assisted translation and quality-aware decoding.

Our solution ranked first among all shared task participants in both translation directions, according to the automatic and human evaluation (Kocmi et al., 2022). The system was only outperformed by human translations, which were anonymously evaluated, the same way as all other submissions. The WMT 2022 organizing committee's certificate, attached as Appendix A, attests to the team's achievement and the results we accomplished.

Development Papers

1. **A Neural Translator Designed to Protect the Eastern Border of the European Union**

Authors:
Artur Nowakowski, Krzysztof Jassem

Venue:
Machine Translation Summit 2021 (Virtual)

Presentation type, date (presenters):
Oral presentation, 18.08.2021 (Artur Nowakowski)

Published paper URL (accessed 16.01.2023):
https://aclanthology.org/2021.mtsummit-up.5.pdf

Author contribution:
Conceptualization and methodology of the research work, implementation of the MT system, conduct of the experiments, integration of lexicons into the MT system, analysis of the automatic evaluation results, writing of the paper.

The paper presents the MT system developed for the Polish State Border Guard. The system is a module of the "Advanced Internet analysis supporting the detection of criminal groups" R&D project (short name: AI Searcher; project no. DOB-BIO9/19/01/2018), which was financed by the Polish National Center for Research and Development. The project took place from 2018 to 2021 and was designed to assist Border Guard officials in searching the Internet for potential criminal activity.

The Border Guard officials use search engines to find potentially criminal content on the Internet. As they also need to find content in foreign languages, including Russian, Ukrainian, and Belarusian, they required an MT system to assist them in this task. The system's task is to translate the input query, which is in Polish, into all of the aforementioned foreign languages. The results found on the Internet are then translated back to Polish for further processing by other modules.

What sets this system apart from other standard MT systems is the use of hand-crafted lexicons. Specific terminology, such as the names of drugs, alcohols and places must be translated accordingly. Another requirement of the Border Guard was that names and surnames of individuals should be transliterated according to their lexicons, rather than translated. Such terminology is difficult to translate using standard MT systems as it does not appear frequently in the training data. To address these issues, we used a method that allows for the incorporation of lexicons into morphologically rich languages, as described in Chapter 2 (Neural Machine Translation with Inflected Lexicon).

The AI Searcher project was successfully deployed at the headquar-

ters of the Polish Border Guard by the end of 2021.

2. **nEYron: Implementation and Deployment of an MT System for a Large Audit & Consulting Corporation**

Authors:
Artur Nowakowski, Krzysztof Jassem, Maciej Lison, Rafał Jaworski, Tomasz Dwojak, Karolina Wiater, Olga Posesor

Venue:
The 23rd Annual Conference of the European Association for Machine Translation (Ghent, Belgium)

Presentation type, date (presenters):
Poster presentation, 03.06.2022 (Artur Nowakowski)

Published paper URL (accessed 16.01.2023):
https://aclanthology.org/2022.eamt-1.21.pdf

Author contribution:
Provision of the MT capabilities of the system, machine translation models training, development of the machine translation service, development of the front-end application, deployment of the final version of the system, writing of the paper.

The paper reports on the implementation and deployment of an MT system at the Polish branch of EY Global Limited. The work on the system was conducted as part of an Industrial PhD program at Poleng.

The system supports CAT (computer-assisted translation) and MT functionalities, such as translation memory fuzzy search, document translation, and post-editing. It also meets less common, customer-specific expectations, such as the preservation of formatting between input and output documents and the calculation of approximate costs of human translation. The system supports translation in two directions: English-Polish and Polish-English.

The paper discusses the challenges encountered during system development and deployment, particularly in relation to the MT model training process, security checks, and installation in the production environment. Development of the system began with a proof-of-concept in 2018 and ended with the deployment of the final version in October 2021.

3. **POLENG MT: An Adaptive MT Platform**

Authors:
Artur Nowakowski, Krzysztof Jassem, Maciej Lison, Kamil Guttmann, Mikołaj Pokrywka

Venue:
The 23rd Annual Conference of the European Association for Machine Translation (Ghent, Belgium)

Presentation type, date (presenters):
Poster presentation, 02.06.2022 (Artur Nowakowski, Kamil Guttmann, Mikołaj Pokrywka)

Published paper URL (accessed 16.01.2023):
https://aclanthology.org/2022.eamt-1.48.pdf

Author contribution:
Training of the MT models, development of the machine translation service, development of the front-end application, deployment of the final version of the system, writing of the paper.

The paper introduces POLENG MT, an MT platform that can be used as a cloud web application or an on-site solution. The development of the platform was conducted as part of an Industrial PhD program at Poleng.

The platform is capable of providing accurate document translation, including the transfer of formatting between input and output documents. It currently supports translation in the following language pairs: Polish-English, Polish-Ukrainian, and Polish-Russian, in both directions.

The main feature of the on-site version is dedicated customer adaptation, which includes training on specialized texts and the application of forced terminology translation to meet the user's needs. It allows for the extraction and incorporation of specialized inflected lexicons in the translation. The method for incorporating inflected lexicons has been described in Chapter 2 (Neural Machine Translation with Inflected Lexicon).

References

Akhbardeh, F., Arkhangorodsky, A., Biesialska, M., Bojar, O., Chatterjee, R., Chaudhary, V., Costa-jussa, M. R., España-Bonet, C., Fan, A., Federmann, C., Freitag, M., Graham, Y., Grundkiewicz, R., Haddow, B., Harter, L., Heafield, K., Homan, C., Huck, M., Amponsah-Kaakyire, K., . . . Zampieri, M. (2021). Findings of the 2021 conference on machine translation (WMT21). *Proceedings of the Sixth Conference on Machine Translation*, 1–88 (cited on page 7).

Freitag, M., Rei, R., Mathur, N., Lo, C.-k., Stewart, C., Avramidis, E., Kocmi, T., Foster, G., Lavie, A., & Martins, A. F. T. (2022). Results of wmt22 metrics shared task: Stop using bleu – neural metrics are better and more robust. *Proceedings of the Seventh Conference on Machine Translation*, 46–68 (cited on pages 1, 5).

Holtzman, A., Buys, J., Du, L., Forbes, M., & Choi, Y. (2020). The curious case of neural text degeneration. *International Conference on Learning Representations* (cited on page 6).

Kocmi, T., Bawden, R., Bojar, O., Dvorkovich, A., Federmann, C., Fishel, M., Gowda, T., Graham, Y., Grundkiewicz, R., Haddow, B., Knowles, R., Koehn, P., Monz, C., Morishita, M., Nagata, M., Nakazawa, T. i., Novák, M., Popel, M., Popović, M., & Shmatova, M. (2022). Findings of the 2022 conference on machine translation (wmt22). *Proceedings of the Seventh Conference on Machine Translation*, 1–45 (cited on page 8).

Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., Lewis, M., & Zettlemoyer, L. (2020). Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, *8*, 726–742. https://doi.org/10.1162/tacl_a_00343 (cited on page 1)

Mroczkowski, R., Rybak, P., Wróblewska, A., & Gawlik, I. (2021). HerBERT: Efficiently pretrained transformer-based language model for Polish. *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, 1–10 (cited on page 6).

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311–318. https://doi.org/10.3115/1073083.1073135 (cited on page 5)

Popović, M. (2015). ChrF: Character n-gram F-score for automatic MT evaluation. *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 392–395. https://doi.org/10.18653/v1/W15-3049 (cited on page 5)

Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2020). COMET: A neural framework for MT evaluation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2685–2702. https://doi.org/10.18653/v1/2020.emnlp-main.213 (cited on page 5)

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, 3104–3112 (cited on page 5).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc. (Cited on page 1).

Wan, Y., Liu, D., Yang, B., Zhang, H., Chen, B., Wong, D., & Chao, L. (2022). UniTE: Unified translation evaluation. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 8117–8127. https://doi.org/10.18653/v1/2022.acl-long.558 (cited on page 5)

Research Papers

Neural Machine Translation with Inflected Lexicon

2

Abstract

The paper presents experiments in Neural Machine Translation with lexical constraints into a morphologically rich language. In particular, we introduce a method, based on constrained decoding, which handles the inflected forms of lexical entries and does not require any modification to the training data or model architecture. To evaluate its effectiveness, we carry out experiments in two different scenarios: general and domain-specific. We compare our method with baseline translation, i.e. translation without lexical constraints, in terms of translation speed and translation quality. To evaluate how well the method handles the constraints, we propose new evaluation metrics which take into account the presence, placement, duplication and inflectional correctness of lexical terms in the output sentence.

2.1 Introduction

The incorporation of an inflected lexicon into Neural Machine Translation (NMT) enables system developers to adapt the translation to specific domains, and users to adjust translations of phrases generated by the translation system.

Phrase-Based Statistical Machine Translation (PB-SMT; Setiawan et al., 2005) provided control over system output, e.g. by using a domain-specific lexicon. The shift from phrase tables in PB-SMT to a continuous-valued representation of text in NMT has made it more difficult to incorporate lexical constraints into the translation process. The task of integrating the lexicon and a neural translator is even more challenging for highly morphological languages, when the lexical items should be correctly inflected in the output text.

We carry out experiments for translation with inflected lexical constraints. As the target language of the translation we choose Polish, whose inflection is typical of the Slavic languages. The number of declination cases is six, and the verbal groups are inflected by tense, number, and person. In terms of correct inflection of the output, translation from English to Polish seems to be a more challenging task than translation in the other direction.

Unlike in some preceding experiments, we require that the lexicon may be modified after the model training has been completed. We believe that in post-editing mode users expect the translation engine to immediately mirror their adjustments to the lexicon.

2.2 Related Work

One of the first papers that addressed the incorporation of a lexicon into an NMT system was Arthur et al., 2016. The authors noticed that NMT

systems tend to produce unexpected output for low-frequency words (such as names of countries). The solution proposed there consisted in designing probability lexicons and combining them with probabilities calculated by an NMT model. Let us note that the motivation for that research was the avoidance of major translation errors, rather than domain adaptation.

Anderson et al., 2017 introduced the concept of a Constrained Beam Search (CBS) in the task of picture captioning. The proposed algorithm forces the inclusion of selected tag words in the output. The solution makes it possible to apply, in the caption, words that were never present in the training data. The method yields the desired results provided that these out-of-vocabulary tags are based on "ground truth", such as labels obtained by reliable object detectors.

The application of CBS for lexical interference in the process of neural text generation was investigated in Hokamp and Liu, 2017. In the decoding phase, the beam is limited only to hypotheses, which include predefined phrases or words. The algorithm called the Grid Beam Search (GBS) may be used for various text-generation tasks where auxiliary knowledge is expected to be incorporated into the text output. If applied to translation, the solution searches for lexical items in the source text and, in positive cases, imposes the presence of their equivalents on the beam.

Hasler et al., 2018 pointed out a danger in the CBS method resulting from the lack of correspondence between constraints and the source words they cover – the placement of the constraint translation in the output may not be correct. To avoid this undesirable effect, the authors "employ alignment information between target-side constraints and their corresponding source words."

The downside of the above algorithms is their complexity: exponential (CBS) or linear (GBS) in the number of constraints. Post and Vilar, 2018 introduce an improvement of the GBS algorithm, called Dynamic Beam Allocation (DBA), which divides the fixed-size beam into "banks": sets of hypotheses that satisfy the same number of constraints. The algorithm depends only on the sentence length and the beam size, being independent of the number of constraints.

Hu et al., 2019 notice that the use of positive (specific tokens must be present in the output) or negative (specific tokens must not be generated) constraints may be useful in rewriting tasks other than translation. Rewriting (see e.g. Napoles et al., 2016) consists in generating an output sentence in the same language and similar in meaning to the input. Examples of such tasks are paraphrasing, question answering and natural language inference. Hu et al., 2019 regard it as crucial to focus on complexity issues to speed up the process of constrained text generation. They develop a "vectorized DBA algorithm with trie representation", which speeds up the computations fivefold compared with the standard DBA algorithm.

Further complexity improvements to constrained NMT are suggested in Song et al., 2019. They apply the idea of so-called "code-switching", which consists in injecting the target terms to the source side of the training data. The idea is similar to that of using placeholder tags to stand for rare names (Luong et al., 2015) or named entities (Deng et al., 2017). The difference is that the direct translations of terms are placed

in the source text instead of tags. The output text is then left untouched. The authors claim that the idea improves translation because it "does not hurt unconstrained words." We believe, however, that in some (not rare) cases the replacement of the constrained word(s) should have an impact on the choice of unconstrained words.

Dinu et al., 2019 apply the idea of "code-switching" in two different scenarios. Depending on the experimental setup the target terms are placed either beside or in place of their source equivalents.

The code-switching method is faster than the previous implementations based on constrained decoding (the presence of constraints need not be verified in the beam). The downside is that it requires interference with the training data.

Exel et al., 2020 verify the efficiency of the code-switching method in an industrial scenario. They inject the terminology of the SAP company into two translation pairs, English–German and English–Russian, and provide both automatic and human evaluation.

From our point of view, the English–Russian case is more interesting because it addresses the problem of inflected forms of lexical constraints. There are two questions of interest to us:

1. How to ensure that the terms are inserted into the target sentence in the correct inflected form?
2. How to evaluate the correctness of term inflection in the translation?

We could not find answers to the above questions in the paper. Therefore, we investigated other solutions, such as the Levenshtein Transformer, introduced in Gu et al., 2019. The method uses "dual policy learning", which consists in using two adversary policies during learning: when training one policy, the output from its adversary at the previous iteration is used as input. In the Levenshtein Transformer the two policies are deletion and insertion of a token in the generated text. The idea is supposed to resemble human intelligence, which sometimes chooses to delete an item from the text intended as output.

In Susanto et al., 2020 the Levenshtein Transformer was used to incorporate lexical constraints in NMT. The idea seemed more appealing to us than code-switching because it does not interfere with the training procedure. However, our initial experiments with the methodology did not succeed – the inflected forms of lexicon entries were not generated correctly. Finally, we decided to carry out our experiments with the base Transformer model, as introduced by Vaswani et al., 2017, and design an algorithm that handles inflected forms of lexical constraints based on the GBS algorithm.

2.3 Experiments

The purpose of our experiments was to find an efficient solution that applies lexical constraints in interactive-mode translation into a morphologically rich language. To be more specific, we aimed to develop a method that would satisfy the following conditions:

- The translation takes into account inflection of lexical items;

- The training data need not be modified.

2.3.1 Evaluation Metrics

We used the standard BLEU metric for translation quality evaluation on the untokenized reference sentences. We also wanted to verify whether the following conditions are satisfied:

1. The target term is present in the output sentence;
2. The target term is properly placed;
3. The target term is not duplicated;
4. The target term is correctly inflected.

Following Exel et al., 2020, we used the Term Rate (TR) to evaluate condition 1. We define Placement Rate (PR) to evaluate condition 2, Duplication Rate (DR) to evaluate condition 3, and Inflection Rate (IR) to evaluate condition 4.

$$TR = \frac{count(terms\ generated\ in\ output)}{count(terms\ that\ appeared\ in\ input)}$$

$$PR = \frac{count(terms\ placed\ properly\ in\ output)}{count(terms\ generated\ \ in\ output)}$$

$$DR = \frac{count(terms\ not\ duplicated\ in\ output)}{count(terms\ generated\ in\ output)}$$

$$IR = \frac{count(terms\ inflected\ properly\ )}{count(terms\ generated\ in\ output)}$$

2.3.2 Lexical Constraints

The lexical constraints were extracted from Paterson, 2015, a compendium of Polish and English accounting forms, available under a Creative Commons license. The number of extracted term pairs was 1197.

We used the Google search engine to obtain inflected forms of Polish terms. Specifically, we queried the search engine with the base forms of terms and scraped snippets from the first 20 pages of query results. We then limited the number of inflected variants to those that covered 95% of cases (we found out that 5% rare cases were more often than not erroneous). The most frequent number of inflected forms for one term was between two and five.

1: http://sgjp.pl

This language-agnostic approach allowed us to obtain the most widely used inflected forms of multi-word phrases, which are not present in Polish vocabularies such as *SGJP*,[1] which only include inflected forms of single words.

2.3.3 Data Preparation

The direction of translation was from English into Polish. The training corpus consisted of the *Europarl v8*, *EUBookshop v2*, *JRC-Acquis v3.0*, *TildeMODEL v2018* and *Wikipedia v1.0* corpora and most of *DGT v2019*. All corpora were downloaded from the *OPUS*[2] collection (Tiedemann, 2012) and filtered using the *Bicleaner*[3] and *Bifixer*[4] (Ramírez-Sánchez et al., 2020) tools. The size of the training corpus after filtering was 3,103,819 segments.

2: https://opus.nlpl.eu/

3: https://github.com/bitextor/bicleaner

4: https://github.com/bitextor/bifixer

For the validation set, we used 2000 sentences from the *DGT* corpus, removing them from the training set.

For the test sets, for two experiments, we extracted respectively 1000 and 1104 segment pairs from the *DGT* corpus, making sure that they did not overlap with either the training set or the validation set. The first test set contained randomly selected segments in which at least one lexical term appeared in the source-side segment, regardless of the presence of target lexical equivalents. We further refer to this experiment as the general scenario. The second test set contained all segments from the corpus in which, for each lexical term in the source-side segment, one of the inflected forms of its lexical equivalent appeared in the target-side segment. We refer to this as the domain-specific scenario.

All of the sets were processed by the BPE algorithm (Sennrich et al., 2016) with the SentencePiece tool[5] (Kudo & Richardson, 2018).

5: https://github.com/google/sentencepiece

2.3.4 Experimental Setup

6: https://github.com/pytorch/fairseq

We carried out our experiments using *fairseq*[6] (Ott et al., 2019), a PyTorch-based open-source sequence modeling toolkit.

We designed a lexicon where for each entry in the source language we provided multiple inflected forms of the corresponding entry in the target language, as described in 2.3.2. In order to use constrained decoding, we trained the Transformer model with a base configuration of six encoding and decoding layers, as introduced by Vaswani et al., 2017.

To obtain translations with correct inflected forms of lexical constraints, we introduced the following algorithm, which applies constrained decoding:

1. Translate the input sentence without any lexical constraints; calculate its average log-likelihood score.
2. Use the fuzzy search (see below) to check whether all lexical constraints are satisfied in the translation; end if the answer is positive.
3. For each unsatisfied lexical constraint:
    a) Take all inflected forms of its lexical equivalent from the lexicon.
    b) For each inflected form:
    Use lexically constrained decoding to translate the input sentence with the inflected form required to be present in the output.

   c) Select the inflected form for which the translation has the highest average log-likelihood score.
4. Use lexically constrained decoding to generate the translation with the list of constraints selected in step 3.
5. Mark the translation as "ok" if the score of the selected translation is not worse than half of the score of the unconstrained translation; otherwise mark it as "warning".

Marking translation output as "warning" allowed us to detect potential errors in the constrained translation (mismatched context, a missing morphological form), thus making it possible to revert to the unconstrained translation if an error was detected.

In the fuzzy search (step 2 of the algorithm) we applied the Token Sort Ratio method, as implemented in the *spaczz*[7] library. The Token Sort Ratio algorithm splits the compared strings into tokens, sorts each list of tokens alphabetically and compares the corresponding elements of the lists using the Levenshtein distance on the level of characters. We considered the found term to match the search term if the similarity ratio, calculated by the algorithm, was not lower than 90%.

7: https://github.com/gandersen101/spaczz

We used a beam size of 5 for decoding in step 3(b) of the above algorithm. We used a beam size of 12 in steps 1 and 4.

2.3.5 Evaluation

The baseline for our solution is the translation without lexical constraints. To assess the effectiveness of our method, we compared it with the baseline in the general and domain-specific scenarios and verified the following aspects of its performance:

1. translation quality (BLEU score);
2. translation speed (measured in seconds);
3. Term Rate;
4. Placement Rate;
5. Duplication Rate;
6. Inflection Rate.

We performed a manual check to calculate the Term Rate, Placement Rate, Duplication Rate and Inflection Rate. The BLEU scores were calculated using the *SacreBLEU*[8] tool (Post, 2018).

8: https://github.com/mjpost/sacrebleu

We calculated separate BLEU scores for the entire test sets and for the set of sentences for which the constrained decoding was actually used (i.e. sentences for which the result of unconstrained translation did not satisfy all of the lexical constraints). Additionally, we calculated the BLEU score for the scenario where "warning" translations are reverted to the unconstrained translations. Manual evaluation metrics were calculated for the entire test sets.

The speed tests were performed on a single NVIDIA RTX 2070 GPU and the AMD Ryzen 7 3700X 8-core processor, using the entire test sets. When translating with the lexicon, the first (unconstrained) and last (with all selected inflected forms) translations were performed with a batch size of 1, while the search for the correct inflected forms was performed as a single batch with the size depending on the number of constraints and

their inflected forms. The time spent on the search for the appearance of lexicon entries was also included. When translating without a lexicon, we used a batch size of 1.

In the tables of results, we refer to the unconstrained translation as *base*, the translation using the lexicon as *lexicon*, and the translation using the lexicon with reversion to the original in case of "warning" as *lexicon-revert*.

Experiment 1: General Scenario

In this scenario the test set consisted of sentences which contained lexical terms in the source text, independently of the presence of their equivalents in the target text.

Constrained decoding was used in the translation of 622 out of 1000 sentences, which corresponds to 62.20% of the entire test set. In these 622 translated sentences, 404 were marked as "ok" and 218 as "warning". In the 378 sentences where constrained decoding was not used, the unconstrained translation satisfied all lexical constraints.

The BLEU results for the experiment are presented in Table 2.1, the manual evaluation results for the *lexicon* translation type are presented in Table 2.2, and translation speed results are presented in Table 2.3.

Translation type	Entire set	Constrained sentences
base	**42.21**	**41.67**
lexicon	39.91	37.59
lexicon-revert	40.97	39.68

Table 2.1: BLEU scores obtained in the general scenario

Metric	Result
Term Rate	98.90
Placement Rate	90.79
Duplication Rate	97.00
Inflection Rate	76.48

Table 2.2: Results of manual evaluation of *lexicon* translation type in the general scenario

Translation type	Time result (s)
base	**273.88**
lexicon	1200.26

Table 2.3: Translation speed in the general scenario

Unsurprisingly, the BLEU results are higher for translation without using the lexicon. This is consistent with the intuition that in the general scenario using the lexicon to correct the neural translation leads to a decrease in the BLEU score. The reversion to the unconstrained translation in situations where the output was marked "warning" may mitigate this effect to some extent. The reversion was particularly helpful in situations where the output from translation with the lexicon was corrupted; for instance, when constraints were placed at the end of the generated sentence or in the wrong inflected form, due to mismatched context or absence of the correct inflected form of the term in the lexicon.

The manual evaluation results indicate that the constraint accuracy in the general scenario is high for three metrics: Term Rate, Placement Rate

and Duplicate Rate. Inflection Rate, however, is rather low because of the missing relevant inflected forms of the terms in the lexicon.

Term Rate is lower than 100% because in a few cases the lexical equivalent was generated in a different inflected form than any of the forms present in the lexicon. This is due to the fact that constraints are also divided into subwords (by the BPE algorithm) before the constrained decoding. In some rare cases this may lead to the proper generation of constraint subword units in the output sentence, but to a different constraint form than is required after the sentence is "de-BPEed".

Translation speed results show that constrained decoding significantly slows down the translation process. The decrease in speed is dependent on the number of constraints and the number of inflected forms of target lexical terms.

Experiment 2: Domain-specific Scenario

In Scenario 2 we evaluated the effectiveness of lexically constrained translation for the sentences where all lexical constraints were satisfied in the reference translation.

Constrained decoding was used in the translation of 150 out of 1104 sentences, which corresponds to 13.59% of the entire test set. In these 150 translated sentences, 143 were marked as "ok" and 7 as "warning". In the 954 sentences where constrained decoding was not used, all lexical constraints were satisfied in the unconstrained translation.

The BLEU results for the experiment are presented in Table 2.4, the manual evaluation results for the *lexicon* translation type are presented in Table 2.5, and translation speed results are presented in Table 2.6.

Table 2.4: BLEU scores obtained in the domain-specific scenario

Translation type	Entire set	Constrained sentences
base	42.30	36.17
lexicon	**42.76**	**39.80**
lexicon-revert	42.73	39.54

Table 2.5: Results of manual evaluation of *lexicon* translation type in the domain-specific scenario

Metric	Result
Term Rate	99.37
Placement Rate	98.37
Duplication Rate	99.09
Inflection Rate	97.28

Table 2.6: Translation speed in the domain-specific scenario

Translation type	Time result (s)
base	**316.79**
lexicon	540.56

The BLEU metric results show that translation with the lexicon leads to an increase in translation quality when the context of the input sentences matches the context of the lexicon and when the relevant inflected forms are present in the lexicon. Reverting to the translation without constraints in situations where the output was marked as "warning" resulted in a

very slight decrease in the BLEU score. This is probably due to the fact that such cases were too rare for the results to be reliable.

The manual evaluation results indicate that our method is very effective in selecting a correct inflected form of the constraint in the domain-specific scenario. All of the metrics returned high scores, including the Inflection Rate.

In this scenario, lexical constraints were not satisfied in the unconstrained translation only in 13.59% of cases. This shows that the neural translation model itself is capable of generating translations with the correct terminology given adequate context. It is concluded that the use of lexical constraints in NMT improves translation quality only in scenarios where the lexicon is highly specific for the translation context.

2.3.6 Examples of Translation with Inflected Lexicon

Table 2.7 shows two examples of sentences translated with and without the use of inflected lexicon. The lexicon entries consist of a term in English language with the equivalent in Polish language along with its comma-separated list of inflectional forms.

Table 2.7: Examples of translation with inflected lexicon

Lexicon entry	audit committee -> komisja rewizyjna, komisji rewizyjnej, komisją rewizyjną, komisję rewizyjną
Source sentence	The **audit committee** should be composed exclusively of non-executive or supervisory directors.
Translation without lexicon	Komitet ds. audytu powinien składać się wyłącznie z dyrektorów niewykonawczych lub będących członkami rady nadzorczej.
Translation with lexicon	W skład **komisji rewizyjnej** powinni wchodzić wyłącznie dyrektorzy niewykonawczy lub będący członkami rady nadzorczej.
Lexicon entry	outlay -> nakład, nakładu, nakłady, nakładów
Source sentence	The statement of the beneficiary's outlay shall be produced in support of any request for a new payment.
Translation without lexicon	Deklarację wydatków beneficjenta przedstawia się na poparcie każdego wniosku o nową płatność.
Translation with lexicon	Deklarację **nakładów** beneficjenta przedstawia się na poparcie każdego wniosku o nową płatność.

2.4 Conclusions

We have examined a new approach to terminology translation into a morphologically rich language with the use of lexicons. We verified that our method, based on constrained decoding, enables the selection of accurate inflected forms of lexical constraints. The method yields an increase in the BLEU metric score provided that appropriate lexical variants of terms are present in the lexicon and the input sentence context is consistent with the lexicon entries. The cost of the algorithm is a decrease in the translation speed. We proposed new metrics for the evaluation of terminology translations: Placement Rate, Duplication Rate and Inflection Rate. The manual evaluation results show that our method

ensures terminological adequacy and consistency when translating into a morphologically rich language in domain-specific scenarios.

2.5 Future Work

We believe that there is still much to explore in the field of terminology translation. In future experiments, we plan to compare our solution with the code-switching approach (Dinu et al., 2019; Song et al., 2019) and to investigate methods which do not have such a negative impact on translation speed as constrained decoding. Another potential direction for improvement is to design a method that does not require the presence of multiple inflected forms in the lexicon before translation.

References

Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2017). Guided open vocabulary image captioning with constrained beam search. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 936–945. https://doi.org/10.18653/v1/D17-1098 (cited on page 16)

Arthur, P., Neubig, G., & Nakamura, S. (2016). Incorporating discrete translation lexicons into neural machine translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1557–1567. https://doi.org/10.18653/v1/D16-1162 (cited on page 15)

Deng, Y., Kim, J., Klein, G., Kobus, C., Segal, N., Servan, C., Wang, B., Zhang, D., Crego, J. M., & Senellart, J. (2017). SYSTRAN purely neural MT engines for WMT2017. In O. Bojar, C. Buck, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, M. Huck, A. Jimeno-Yepes, P. Koehn, & J. Kreutzer (Eds.), *Proceedings of the second conference on machine translation, WMT 2017, copenhagen, denmark, september 7-8, 2017* (pp. 265–270). Association for Computational Linguistics. https://doi.org/10.18653/v1/w17-4722. (Cited on page 16)

Dinu, G., Mathur, P., Federico, M., & Al-Onaizan, Y. (2019). Training neural machine translation to apply terminology constraints. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3063–3068. https://doi.org/10.18653/v1/P19-1294 (cited on pages 17, 24)

Exel, M., Buschbeck, B., Brandt, L., & Doneva, S. (2020). Terminology-constrained neural machine translation at SAP. *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, 271–280 (cited on pages 17, 18).

Gu, J., Wang, C., & Zhao, J. (2019). Levenshtein transformer. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc. (Cited on page 17).

Hasler, E., de Gispert, A., Iglesias, G., & Byrne, B. (2018). Neural machine translation decoding with terminology constraints. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 506–512. https://doi.org/10.18653/v1/N18-2081 (cited on page 16)

Hokamp, C., & Liu, Q. (2017). Lexically constrained decoding for sequence generation using grid beam search. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1535–1546. https://doi.org/10.18653/v1/P17-1141 (cited on page 16)

Hu, J. E., Khayrallah, H., Culkin, R., Xia, P., Chen, T., Post, M., & Van Durme, B. (2019). Improved lexically constrained decoding for translation and monolingual rewriting. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 839–850. https://doi.org/10.18653/v1/N19-1090 (cited on page 16)

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71. https://doi.org/10.18653/v1/D18-2012 (cited on page 19)

Luong, T., Sutskever, I., Le, Q., Vinyals, O., & Zaremba, W. (2015). Addressing the rare word problem in neural machine translation. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 11–19. https://doi.org/10.3115/v1/P15-1002 (cited on page 16)

Napoles, C., Callison-Burch, C., & Post, M. (2016). Sentential paraphrasing as black-box machine translation. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, 62–66. https://doi.org/10.18653/v1/N16-3013 (cited on page 16)

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., & Auli, M. (2019). Fairseq: A fast, extensible toolkit for sequence modeling. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 48–53. https://doi.org/10.18653/v1/N19-4009 (cited on page 19)

Paterson, R. (2015). *Compendium of Acounting in Polish & English* (cited on page 18).

Post, M. (2018). A call for clarity in reporting BLEU scores. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 186–191 (cited on page 20).

Post, M., & Vilar, D. (2018). Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1314–1324. https://doi.org/10.18653/v1/N18-1119 (cited on page 16)

Ramírez-Sánchez, G., Zaragoza-Bernabeu, J., Bañón, M., & Ortiz-Rojas, S. (2020). Bifixer and bicleaner: Two open-source tools to clean

your parallel data. *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, 291–298 (cited on page 19).

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1715–1725. https://doi.org/10.18653/v1/P16-1162 (cited on page 19)

Setiawan, H., Li, H., Zhang, M., & Ooi, B. C. (2005). Phrase-based statistical machine translation: A level of detail approach. *Second International Joint Conference on Natural Language Processing: Full Papers*. https://doi.org/10.1007/11562214_51 (cited on page 15)

Song, K., Zhang, Y., Yu, H., Luo, W., Wang, K., & Zhang, M. (2019). Code-switching for enhancing NMT with pre-specified translation. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 449–459. https://doi.org/10.18653/v1/N19-1044 (cited on pages 16, 24)

Susanto, R. H., Chollampatt, S., & Tan, L. (2020). Lexically constrained neural machine translation with Levenshtein transformer. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 3536–3543. https://doi.org/10.18653/v1/2020.acl-main.325 (cited on page 17)

Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, 2214–2218 (cited on page 19).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc. (Cited on pages 17, 19).

Approaching English-Polish Machine Translation Quality Assessment with Neural-based Methods

3

Abstract

This paper presents our contribution to the PolEval 2021 Task 2: *Evaluation of translation quality assessment metrics*. We describe experiments with pre-trained language models and state-of-the-art frameworks for translation quality assessment in both *nonblind* and *blind* versions of the task. Our solutions ranked second in the *nonblind* version and third in the *blind* version.

3.1 Introduction

Machine translation quality evaluation is the task of assessing translation quality based on a reference translation. In the past, traditional machine translation evaluation metrics such as BLEU (Papineni et al., 2002), METEOR (Banerjee & Lavie, 2005), or CHRF (Popović, 2015) relied on lexical-level features between the machine translation hypothesis and the reference translation. They remain popular to this day due to their computational speed and the fact that they can be applied to any translation direction.

The rise of Neural Machine Translation (NMT) in recent years has shown that high-quality NMT systems are often mistreated by lexical-level evaluation metrics, as such systems can generate correct translation that is lexically distant from a reference translation.

Recent advances in the field of neural language modeling (Conneau et al., 2020; Devlin et al., 2019) led to the creation of BERT cosine similarity-based metrics, such as BERTSCORE (Zhang* et al., 2020), as well as metrics trained on human judgments, such as COMET (Rei et al., 2020) and BLEURT (Sellam et al., 2020). Human judgments include manually assigned quality scores, such as *Direct Assessment* (DA) (Graham et al., 2013), but may also be derived from post-edited translation to calculate post-editing effort in the form of *Human-mediated Translation Edit Rate* (HTER) (Snover et al., 2006).

Machine translation quality estimation (QE) is a different task than evaluation, as the goal is to predict machine translation quality without access to a reference translation. Research on QE in recent years has shown that it is possible to achieve high levels of correlation with human judgments based only on a source segment and a machine translation hypothesis (Specia et al., 2020). Existing state-of-the-art frameworks for QE include COMET (Rei et al., 2020), which allows QE models to be trained in a reference-free mode and TRANSQUEST (Ranasinghe et al., 2020), which proposes two new architectures for QE: MONOTRANSQUEST and SIAMESETRANSQUEST.

3.2 Task Description

The goal of Task 2 is to investigate metrics for automatic evaluation of machine translation in the English-Polish translation direction.

The organizers prepared distinct datasets for *nonblind* and *blind* versions of the task. The *nonblind* dataset consists of the following data: source segment, machine translation hypothesis, reference translation, and quality score. The *blind* dataset consists only of machine translation hypothesis and its quality score. The segment quality scores were created by averaging the scores assigned by six human annotators. Unlike most of the current human judgment-based QE tasks, where scores are assigned on a continuous scale (Graham et al., 2013), the task utilizes a standard Likert scale allowing ratings from 1 to 5. The evaluation metric used in both versions of the task is Pearson's *r* correlation score.

The datasets were split into a development set ("dev-0") and two test sets ("test-A" and "test-B"). The first of the test sets ("test-A") was the main test set during the initial testing phase of the competition and was converted to the development set with the release of the final test set ("test-B").

Table 3.1 presents statistics of the provided datasets: the number of segments, the average number of source tokens, the average number of MT hypothesis tokens, the minimum segment quality score, and the average segment quality score.

Table 3.1: Statistics of datasets provided by organizers.

	Nonblind			**Blind**		
	Dev-0	**Test-A**	**Test-B**	**Dev-0**	**Test-A**	**Test-B**
Segments	485	500	1000	485	500	1000
Avg. tokens (source)	18.22	17.36	17.73	-	-	-
Avg. tokens (MT hypothesis)	16.23	15.49	15.78	17.55	16.49	16.57
Min. score	3.0	2.58	2.92	3.08	2.67	2.0
Avg. score	4.30	4.37	4.38	4.33	4.31	4.40

3.3 Solutions

3.3.1 *Nonblind* Task Version Solution

Our final solution to the *nonblind* version of the task is based on COMET. We used the "test-A" dataset as the training data and the "dev-0" dataset as the development data.

COMET uses pre-trained language model as the encoder for the source segment, the machine translation hypothesis, and the reference translation, which are independently encoded. Therefore, we decided to use $\text{HerBERT}_{\text{LARGE}}$ (Mroczkowski et al., 2021) as the pre-trained encoder model. We also experimented with XLM-RoBERTa (Conneau et al., 2020) (XLM-R) as the pre-trained encoder model, but the results were subpar. It is because $\text{HerBERT}_{\text{LARGE}}$ model was trained specifically for the Polish language and initialized with XLM-RoBERTa weights.

We applied gradual unfreezing and discriminative learning rates (Howard & Ruder, 2018), meaning that we kept the encoder model frozen for 8

epochs while the feed-forward regressor was optimized with the learning rate of 3e−5. After 8 epochs, the entire model is fine-tuned but the learning rate is reduced to 1e−5 to avoid catastrophic forgetting. All hyperparameters used for training COMET models are presented in Table 3.4.

We experimented with other state-of-the-art methods for machine translation evaluation as well. We used BERTSCORE with contextual embeddings from the $\text{HerBERT}_{\text{LARGE}}$ model and found that it generates promising results given that it is based on cosine similarity and is not fine-tuned on the task data in any way.

Out of the trained metrics, we also experimented with BLEURT and TRANSQUEST with MONOTRANSQUEST architecture. The BLEURT model was fine-tuned on the open-source *bleurt-base-128* model[1] with default hyperparameters. The TRANSQUEST model was fine-tuned on the open-source English-to-Any model pre-trained on DA[2] with default hyperparameters. TRANSQUEST is trained only on the source segment and the machine translation hypothesis and does not take into account the reference translation. The final results of all methods used in the *nonblind* version of the task are presented in Table 3.2.

1: https://github.com/google-research/bleurt/blob/master/checkpoints.md

2: https://tharindu.co.uk/TransQuest/models/sentence_level_pretrained

Table 3.2: Results of the *nonblind* version of the task on the "test-B" dataset.

Method	Pearson's r
COMET (HerBERT)	**57.28**
COMET (XLM-R)	53.84
BLEURT	57.25
TRANSQUEST	55.70
BERTSCORE	48.74

3.3.2 *Blind* Task Version Solution

Our final solution to the *blind* version of the task is based on COMET as well.

The provided dataset contains only machine translation hypotheses in this scenario. Therefore, we decided to create synthetic source segments by back-translating the provided machine translation hypotheses into English by using the open-source OPUS-MT (Tiedemann & Thottingal, 2020) NMT model[3], which is based on the Marian (Junczys-Dowmunt et al., 2018) framework.

3: https://huggingface.co/Helsinki-NLP/opus-mt-pl-en

We combined all the data from the *nonblind* dataset with the back-translated data from the *blind* dataset. Then, we randomly selected 100 segment pairs as the development set.

The model training procedure is the same as in the *nonblind* solution. The only difference is that the COMET model was trained in the reference-free mode in this scenario. Hyperparameters used for the *blind* model training are presented in Table 3.4.

In this version of the task, we also conducted experiments using TRANSQUEST. TRANSQUEST model architecture, hyperparameters, and used pre-trained model were the same as in the solution to the *nonblind* version of the task. The final results of all methods used in the *blind* version of the task are presented in Table 3.3.

Table 3.3: Results of the *blind* version of the task on the "test-B" dataset.

Method	Pearson's r
COMET (HerBERT)	**47.93**
COMET (XLM-R)	43.52
TRANSQUEST	41.71

Table 3.4: Hyperparameters used for training COMET models.

Hyperparameter	Nonblind model	Blind model
Pre-trained encoder model	$\text{HerBERT}_{\text{LARGE}}$	$\text{HerBERT}_{\text{LARGE}}$
Optimizer	Adam (default parameters)	AdamW (default parameters)
Learning rate	3e−5 and 1e−5	3.1e−5 and 1e−5
Layer-wise decay	-	0.95
Num. of frozen epochs	8	0.3
Batch size	4	2
Accumulated gradient batches	2	4
Loss function	MSE	MSE
Dropout	0.15	0.15
Feed-forward hidden units	4096, 2048	2048, 1024
Feed-forward activation function	Tanh	Tanh

3.4 Conclusions

We presented our contribution to the PolEval 2021 Task 2: *Evaluation of translation quality assessment metrics*.

The experiments consisted in comparing state-of-the-art methods for translation quality assessment in the English-Polish translation direction. The final solutions are based on the COMET framework. The solutions achieved second and third place in the *nonblind* and *blind* versions of the task, respectively. In the *blind* version of the task, we presented a procedure for creating a synthetic source segment input by back-translating machine translation hypothesis. All of the described methods are also worth further investigation in future experiments, as they generate competitive results.

The code and models used for creating the solutions are open-source and available on GitHub[4].

4: https://github.com/arturnn/poleval2021-qe

References

Banerjee, S., & Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 65–72 (cited on page 27).

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. (Cited on pages 27, 28).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. https://doi.org/10.18653/v1/N19-1423 (cited on page 27)

Graham, Y., Baldwin, T., Moffat, A., & Zobel, J. (2013). Continuous measurement scales in human evaluation of machine translation. *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, 33–41 (cited on pages 27, 28).

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 328–339. https://doi.org/10.18653/v1/P18-1031 (cited on page 28)

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Fikri Aji, A., Bogoychev, N., Martins, A. F. T., & Birch, A. (2018). Marian: Fast neural machine translation in C++. *Proceedings of ACL 2018, System Demonstrations*, 116–121 (cited on page 29).

Mroczkowski, R., Rybak, P., Wróblewska, A., & Gawlik, I. (2021). HerBERT: Efficiently pretrained transformer-based language model for Polish. *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, 1–10 (cited on page 28).

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311–318. https://doi.org/10.3115/1073083.1073135 (cited on page 27)

Popović, M. (2015). ChrF: Character n-gram F-score for automatic MT evaluation. *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 392–395. https://doi.org/10.18653/v1/W15-3049 (cited on page 27)

Ranasinghe, T., Orasan, C., & Mitkov, R. (2020). Transquest: Translation quality estimation with cross-lingual transformers. *Proceedings of the 28th International Conference on Computational Linguistics* (cited on page 27).

Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2020). COMET: A neural framework for MT evaluation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2685–2702. https://doi.org/10.18653/v1/2020.emnlp-main.213 (cited on page 27)

Sellam, T., Das, D., & Parikh, A. (2020). BLEURT: Learning robust metrics for text generation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7881–7892. https://doi.org/10.18653/v1/2020.acl-main.704 (cited on page 27)

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., & Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, 223–231 (cited on page 27).

Specia, L., Blain, F., Fomicheva, M., Fonseca, E., Chaudhary, V., Guzmán, F., & Martins, A. F. T. (2020). Findings of the WMT 2020 shared task on quality estimation. *Proceedings of the Fifth Conference on Machine Translation*, 743–764 (cited on page 27).

Tiedemann, J., & Thottingal, S. (2020). OPUS-MT — Building open translation services for the World. *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)* (cited on page 29).

Zhang*, T., Kishore*, V., Wu*, F., Weinberger, K. Q., & Artzi, Y. (2020). Bertscore: Evaluating text generation with bert. *International Conference on Learning Representations* (cited on page 27).

Adam Mickiewicz University's English-Hausa Submissions to the WMT 2021 News Translation Task

4

Abstract

This paper presents the Adam Mickiewicz University's (AMU) submissions to the WMT 2021 News Translation Task. The submissions focus on the English↔Hausa translation directions, which is a low-resource translation scenario between distant languages. Our approach involves thorough data cleaning, transfer learning using a high-resource language pair, iterative training, and utilization of monolingual data via back-translation. We experiment with NMT and PB-SMT approaches alike, using the base Transformer architecture for all of the NMT models while utilizing PB-SMT systems as comparable baseline solutions.

4.1 Introduction

We describe the Adam Mickiewicz University's submissions to the WMT 2021 News Translation Task. We focused on translation between Hausa and English – a low-resource translation scenario between distant languages. Our methods combine data cleaning with OpusFilter (Aulamo et al., 2020) and fastText (Joulin et al., 2016), transfer learning (Aji et al., 2020; Zoph et al., 2016), iterative training, and back-translation (Sennrich et al., 2016a).

All NMT models were trained with FAIRSEQ (Ott et al., 2019), while the first iteration of the back-translation was generated with Moses (Koehn et al., 2007).

The results presented in the paper are based on the first released development set ("Dev-1"), which consists of 1000 sentences, the final development set ("Dev-full"), which adds additional 1000 sentences to the first development set, and the released test set without additional test suites ("Test"). The test set consists of 1000 sentences in English→Hausa direction and 997 sentences in Hausa→English direction.

The final submissions significantly outperform the vanilla NMT baselines in terms of BLEU (Papineni et al., 2002) metric results, as implemented in SACREBLEU (Post, 2018) with default settings.

All systems were trained in a constrained scenario i.e., using the data provided by the organizers of WMT 2021 only.

4.2 Data Preparation

The quality of the training data has a great impact on the final performance of the NMT models (Rikters, 2018). The data preparation consisted of data cleaning and filtering performed by using OpusFilter (Aulamo et al., 2020) pipelines. We specified separate pipelines for monolingual and parallel data. Data cleaning phase consisted of normalizing punctuation,

removing non-printable characters, and decoding HTML entities by using Moses (Koehn et al., 2007) pre-processing scripts.

We applied subword segmentation on filtered data by using SentencePiece (Kudo & Richardson, 2018) tool with byte-pair-encoding (BPE) (Sennrich et al., 2016b) algorithm. The corpora we used for model training, along with the number of sentences before filtering, are specified in Table 4.1. Number of sentences after filtering is presented in Table 4.2.

Table 4.1: Corpora statistics before filtering.

Data type	Sentences	Corpora
Parallel en-ha	751,560	Khamenei, Opus, ParaCrawl
Monolingual en	41,428,626	News crawl (only 2020)
Monolingual ha	2,311,959	News crawl, CommonCrawl
Parallel de-en	8,600,361	Tilde Rapid, CommonCrawl, Europarl, News commentary, ParaCrawl

Monolingual data filtering For the monolingual data filtering, we defined an OpusFilter pipeline that consists of the following filters:

- deduplication filter,
- sentence length filter,
- word length filter,
- Latin character score filter,
- language identification filter.

The sentence length filter requires that the sentence contain a minimum of 3 and a maximum of 100 words. A maximum of 40 characters is required for the word length. The required Latin character score for a sentence is set to 100%. Language identification filter is based on a fastText (Joulin et al., 2016) language identifier. The open-source fastText language identification models do not identify Hausa, so we used the JW300 corpus from the English-Hausa Opus collection to train our custom language identifier. A sentence must pass all filters to be included in the training data.

Data type	Sentences
Monolingual en	39,812,834
Monolingual ha	1,227,921
Parallel ha-en	494,246

Table 4.2: Monolingual corpora statistics after filtering.

Parallel data filtering The filters used in the parallel data filtering pipeline are nearly identical to those used in the monolingual data filtering pipeline. Filters are applied to both the source and target sentences in this scenario. We also included a length ratio filter with a threshold of 2, indicating that a sentence on the source side can be up to twice as long as a sentence on the target side and vice versa.

A similar pipeline was applied to the German-English data that was used for transfer learning. We downsampled 3M sentence pairs from ParaCrawl due to the imbalance in the German-English data.

4.3 Approach

Our models combine transfer learning from a high-resource language pair (German-English), iterative training, and back-translation. We used FAIRSEQ (Ott et al., 2019) toolkit in our experiments with NMT models,

while we used Moses (Koehn et al., 2007) toolkit for our experiments with PB-SMT models.

All of our NMT models follow the base Transformer architecture (Vaswani et al., 2017), using ReLU as the activation function and Adam (Kingma & Ba, 2015) as the optimizer with the following parameters: $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 1e{-}8$. We set the inverse square root learning rate scheduling with a peak value of 1e−3. We used learning rate warmup stage for 4000 updates with initial learning rate of 1e−7. Dropout probability was set to 0.2, while the attention dropout probability was set to 0.1. We also used label smoothing with a value of 0.1. In the case of baseline English-Hausa models, the joint vocabulary was based on both English and Hausa data. In all cases, the vocabulary size was set to 32,000.

The PB-SMT models were trained with default settings with Moses (Koehn et al., 2007) toolkit. In addition, we trained a 5-gram Operation Sequence Model (Durrani et al., 2013). All language models are 5-gram models and were binarized with KenLM (Heafield et al., 2013). The models were trained on tokenized, word-level, lowercased sentences. Re-casing was applied to the model outputs. After training the base models, we also applied MERT (Minimum Error Rate Training) (Bertoldi et al., 2009; Och, 2003) tuning on the development set.

4.3.1 Baseline Systems

We decided to train baseline models of two types: vanilla Transformer (base) and PB-SMT. The experiments conducted on the first release of the development set showed that PB-SMT performs significantly better than NMT: we achieved +1.8 BLEU score on Hausa→English and +0.7 on English→Hausa. Based on these results, we decided to use PB-SMT models to generate data for the first iteration of iterative training.

When the test set was published, we computed the scores for the baselines. To our surprise, the scores obtained by NMT are much higher than PB-SMT, especially in the Hausa→English direction.

System	HA → EN		EN → HA	
	Dev-1	Test	Dev-1	Test
NMT baseline	12.21	11.44	10.28	11.05
PB-SMT baseline	14.00	6.59	11.02	9.36

Table 4.3: Baseline results according to the automatic evaluation with BLEU metric.

4.3.2 Transfer Learning

According to recent studies, transfer learning (TL) enhances translation quality in low-resource scenarios (Aji et al., 2020; Zoph et al., 2016). We chose the German→English translation direction as a base. In general, we followed (Nguyen & Chiang, 2017) and trained a shared Hausa-German-English vocabulary (BPE). Then, we trained a German→English model using parallel data from the WMT 2021 Translation Task, which was filtered similarly to Hausa-English data. Finally, we used the Hausa-English data to fine-tune the pre-trained German→English model. We obtained a BLEU score of 13.31 on the "Dev-1" development set (+1.1

BLEU compared to the NMT baseline), which was lower than the PB-SMT baseline.

4.3.3 Iterative Back-Translation

Monolingual data has been widely employed in MT to enrich parallel corpora with synthetic data to improve the quality of MT systems, particularly in low-resource scenarios (Bertoldi & Federico, 2009; Bojar & Tamchyna, 2011). We applied the back-translation technique (Edunov et al., 2018) iteratively (Hoang et al., 2018) to translate Hausa and English monolingual data into the other language, using intermediate models to generate incrementally better translations.

1. First, we used the best baseline model (PB-SMT based on Moses) in English→Hausa direction to translate 5M English sentences into Hausa.
2. We used this additional data to train the Hausa→English model by applying transfer learning from the German→English model. We upsampled the original parallel data 10 times to match the size of the back-translated data. We used the resulting NMT model to translate all Hausa monolingual data into English via sampling.
3. We combined the obtained back-translated data with the original parallel corpora to train the English→Hausa model in a manner similar to step 2, with the exception that we did not upsample the parallel data in this scenario due to the fact that back-translated data was generated through sampling.
4. This technique was applied iteratively, resulting in the systems shown in Table 4.4. In all Hausa→English systems except the last, we utilized 5M English monolingual sentences in the model training; in the last system, we used 25M sentences. We used all accessible Hausa monolingual data in all English→Hausa systems.

Table 4.4: Iterative back-translation results of the NMT systems on the "Dev-1" development set according to the automatic evaluation with BLEU metric.

System	HA → EN	EN → HA
1	16.22	-
2	-	13.04
3	20.05	-
4	-	14.38
5	22.85	-
6	-	14.77

4.4 Final Results

Table 4.5 presents the final results for both the English→Hausa and Hausa→English translation directions for both the development and test sets. These results were produced by the final models from the iterative back-translation step described in section 4.3.3.

Table 4.5: Final results according to the automatic evaluation with BLEU metric.

Direction	Dev-1	Dev-full	Test
EN → HA	14.77	21.21	16.15
HA → EN	22.85	25.23	14.13

We notice a severe decrease in BLEU metric results on the test set as compared to the development set, particularly in the Hausa→English direction. This could suggest a domain shift between the two sets. Because our models are heavily based on the back-translated data, some vocabulary, especially proper names, may be missing from the training data.

4.5 Post-submission Work

Due to a lack of computing power and time, our experiments and submissions were based on single model training. After the submission deadline, we retrained the final models three times with different seeds. Table 4.6 presents the results for the ensemble of four models in both directions. We obtained slight improvements on both test sets, but the differences are insignificant. On the other hand, the ensemble performed worse on the development set, especially on the first version.

Direction	Dev-1	Dev-full	Test
EN → HA	14.68	21.00	16.34
HA → EN	21.24	26.25	14.87

Table 4.6: Post-submission models ensemble results according to the automatic evaluation with BLEU metric.

References

Aji, A. F., Bogoychev, N., Heafield, K., & Sennrich, R. (2020). In neural machine translation, what does transfer learning transfer? *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7701–7710. https://doi.org/10.18653/v1/2020.acl-main.688 (cited on pages 33, 35)

Aulamo, M., Virpioja, S., & Tiedemann, J. (2020). OpusFilter: A configurable parallel corpus filtering toolbox. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 150–156. https://doi.org/10.18653/v1/2020.acl-demos.20 (cited on page 33)

Bertoldi, N., & Federico, M. (2009). Domain adaptation for statistical machine translation with monolingual resources. *Proceedings of the Fourth Workshop on Statistical Machine Translation*, 182–189 (cited on page 36).

Bertoldi, N., Haddow, B., & Fouet, J.-B. (2009). Improved minimum error rate training in moses. *The Prague Bulletin of Mathematical Linguistics*, *91*, 7–16. https://doi.org/10.2478/v10108-009-0011-9 (cited on page 35)

Bojar, O., & Tamchyna, A. (2011). Improving translation model by monolingual data. *Proceedings of the Sixth Workshop on Statistical Machine Translation*, 330–336 (cited on page 36).

Durrani, N., Fraser, A., Schmid, H., Hoang, H., & Koehn, P. (2013). Can Markov models over minimal translation units help phrase-based SMT? *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 399–405 (cited on page 35).

Edunov, S., Ott, M., Auli, M., & Grangier, D. (2018). Understanding back-translation at scale. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 489–500. https://doi.org/10.18653/v1/D18-1045 (cited on page 36)

Heafield, K., Pouzyrevsky, I., Clark, J. H., & Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 690–696 (cited on page 35).

Hoang, V. C. D., Koehn, P., Haffari, G., & Cohn, T. (2018). Iterative back-translation for neural machine translation. *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, 18–24. https://doi.org/10.18653/v1/W18-2703 (cited on page 36)

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759* (cited on pages 33, 34).

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In Y. Bengio & Y. LeCun (Eds.), *3rd international conference on learning representations, ICLR 2015, san diego, ca, usa, may 7-9, 2015, conference track proceedings*. (Cited on page 35).

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., & Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, 177–180 (cited on pages 33–35).

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71. https://doi.org/10.18653/v1/D18-2012 (cited on page 34)

Nguyen, T. Q., & Chiang, D. (2017). Transfer learning across low-resource, related languages for neural machine translation. *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 296–301 (cited on page 35).

Och, F. J. (2003). Minimum error rate training in statistical machine translation. *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, 160–167. https://doi.org/10.3115/1075096.1075117 (cited on page 35)

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., & Auli, M. (2019). Fairseq: A fast, extensible toolkit for sequence modeling. *Proceedings of NAACL-HLT 2019: Demonstrations* (cited on pages 33, 34).

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 311–318. https://doi.org/10.3115/1073083.1073135 (cited on page 33)

Post, M. (2018). A call for clarity in reporting BLEU scores. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 186–191 (cited on page 33).

Rikters, M. (2018). Impact of Corpora Quality on Neural Machine Translation. *In Proceedings of the 8th Conference Human Language Technologies - The Baltic Perspective (Baltic HLT 2018)* (cited on page 33).

Sennrich, R., Haddow, B., & Birch, A. (2016a). Improving neural machine translation models with monolingual data. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 86–96. https://doi.org/10.18653/v1/P16-1009 (cited on page 33)

Sennrich, R., Haddow, B., & Birch, A. (2016b). Neural machine translation of rare words with subword units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1715–1725. https://doi.org/10.18653/v1/P16-1162 (cited on page 34)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000–6010 (cited on page 35).

Zoph, B., Yuret, D., May, J., & Knight, K. (2016). Transfer learning for low-resource neural machine translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1568–1575. https://doi.org/10.18653/v1/D16-1163 (cited on pages 33, 35)

Adam Mickiewicz University at WMT 2022: NER-Assisted and Quality-Aware Neural Machine Translation

5

Abstract

This paper presents Adam Mickiewicz University's (AMU) submissions to the constrained track of the WMT 2022 General MT Task. We participated in the Ukrainian ↔ Czech translation directions. The systems are a weighted ensemble of four models based on the Transformer (big) architecture. The models use source factors to utilize the information about named entities present in the input. Each of the models in the ensemble was trained using only the data provided by the shared task organizers. A noisy back-translation technique was used to augment the training corpora. One of the models in the ensemble is a document-level model, trained on parallel and synthetic longer sequences. During the sentence-level decoding process, the ensemble generated the n-best list. The n-best list was merged with the n-best list generated by a single document-level model which translated multiple sentences at a time. Finally, existing quality estimation models and minimum Bayes risk decoding were used to rerank the n-best list so that the best hypothesis was chosen according to the COMET evaluation metric. According to the automatic evaluation results, our systems rank first in both translation directions.

5.1 Introduction

We describe Adam Mickiewicz University's submissions to the constrained track of the WMT 2022 General MT Task. We participated in the Ukrainian ↔ Czech translation directions – a low-resource translation scenario between closely related languages.

The data provided by the shared task organizers was thoroughly cleaned and filtered, as described in section 5.2.

The approach described in section 5.3 is based on combining various MT enhancement methods, including transfer learning from a high-resource language pair (Aji et al., 2020; Zoph et al., 2016), noisy back-translation (Edunov et al., 2018), NER-assisted translation (Modrzejewski et al., 2020), document-level translation, model ensembling, quality-aware decoding (Fernandes et al., 2022), and on-the-fly domain adaptation (Farajian et al., 2017).

The results leading to the final submissions are presented in section 5.4. Additionally, we performed a statistical significance test with paired bootstrap resampling (Koehn, 2004), comparing the baseline solution with the final submission on the test set reference translations released by the shared task organizers. According to the automatic evaluation results based on COMET (Rei et al., 2020) scores, our systems rank first in both translation directions.

Table 5.1: Statistics of the total available corpora and the corpora used for system training after filtering.

Data type		Sentences	Corpora
Monolingual cs	available	448,528,116	News crawl, Europarl v10, News Commentary, Common Crawl, Extended Common Crawl, Leipzig Corpora
	used	59,999,553	
Monolingual uk	available	70,526,415	News crawl, UberText Corpus, Leipzig Corpora, Legal Ukrainian
	used	59,152,329	
Parallel cs-uk	available	12,630,806	OPUS, WikiMatrix, ELRC – EU acts in Ukrainian
	used	8,623,440	

5.2 Data

In the initial stage of system preparation, the sentence-level data was cleaned and filtered using the OpusFilter (Aulamo et al., 2020) toolkit. With the use of the toolkit, language detection filtering based on fastText (Joulin et al., 2016) was performed, duplicates were removed, and heuristics based on sentence length were applied. In particular, we removed sentence pairs with a length ratio over 3 and long sentences (> 200 words). Then, using Moses (Koehn et al., 2007) pre-processing scripts, punctuation was normalized and non-printing characters removed. Finally, the text was tokenized into subword units using SentencePiece (Kudo & Richardson, 2018) with the unigram language model algorithm (Kudo, 2018). For Ukrainian→Czech and Czech→Ukrainian models trained from scratch, we used separate vocabularies for the source and the target language. Each vocabulary consisted of 32,000 units.

We used concatenated data from the Flores-101 (Goyal et al., 2022) benchmark (flores101-dev, flores101-devtest) for our development set, as provided by the task organizers.

Table 5.1 shows statistics for the total available corpora in the constrained track and the corpora used for system training after filtering.

5.3 Approach

We used the Marian (Junczys-Dowmunt et al., 2018) toolkit for all of our experiments. Our model architecture follows the Transformer (big) (Vaswani et al., 2017) settings. For all model training, we used 4x NVIDIA A100 80GB GPUs.

5.3.1 Transfer Learning

For our initial experiments, we used transfer learning (Aji et al., 2020; Zoph et al., 2016) from the high-resource Czech→English language pair. We used only the parallel data provided by the organizers to train the model in this direction. In this case, we created a single joint vocabulary for three languages (Czech, English, Ukrainian), consisting of 32,000 units. The Czech→English model was fine-tuned for the Ukrainian→Czech and Czech→Ukrainian language directions. Our later experiments showed that there were no gains in translation quality compared with models trained from scratch using separate vocabularies for source and target

languages – the upside was that the models took less time to converge during training.

5.3.2 Noisy Back-Translation

We used models created by the transfer learning approach to produce synthetic training data through noisy back-translation (Edunov et al., 2018). Specifically, we applied Gumbel noise to the output layer and sampled from the full model distribution. We used monolingual data available in the constrained track, which included all ~59M Ukrainian sentences after filtering and ~60M randomly selected Czech sentences.

After training the model with concatenated parallel and back-translated corpora, we replaced the training data with filtered parallel data and further fine-tuned the model. We kept the same settings as in the first training pass, training the model until it converged on the development set.

5.3.3 NER-Assisted Translation

Translation in domains such as news, social or conversational texts, and e-commerce is a specialized task, involving such challenges as localization, product names, and mentions of people or events in the content of documents. In such a case, it proved helpful to use off-the-shelf solutions for recognizing named entities. For Czech, the Slavic BERT model (Arkhipov et al., 2019) was used, with which entities such as persons (PER), locations (LOC), organizations (ORG), products (PRO), and events (EVT) were tagged. Due to the lack of support for the Ukrainian language in the Slavic BERT model, the Stanza Named Entity Recognition module (Qi et al., 2020) was used to detect entities in the Ukrainian text, recognizing persons (PER), locations (LOC), organizations (ORG), and miscellaneous items (MISC). With these ready-made solutions, the parallel and back-translated corpora were tagged. The named entity categories were then numbered to assign appropriate source factors to words in the text, supporting the translation process. The source factors were later transferred to subwords in a trivial way.

Source factors (Sennrich & Haddow, 2016) have previously been used to take into account various characteristics of words during the translation process. For example, morphological information, part-of-speech tags, and syntactic dependencies have been added as input to neural machine translation systems to improve the translation quality.

In the same way, it is possible to add information about named entities found in the text (Modrzejewski et al., 2020), making it easier for the model to translate them correctly. However, the AMU machine translation system does not distinguish between inside-outside-beginning (IOB) tags (Ramshaw & Marcus, 1995), treating the named entity tag names as a whole. Specifically, we introduce the following source factors:

- p0: source factor denoting a normal token,
- p1: source factor denoting the PER category,
- p2: source factor denoting the LOC category,
- p3: source factor denoting the ORG category,

- p4: source factor denoting the MISC category,
- p5: source factor denoting the PRO category,
- p6: source factor denoting the EVT category.

An example of a tagged sentence is shown in Figure 5.1.

Models were trained in two settings: concatenation and sum. In the first setting, the factor embedding had a size of 16 and was concatenated with the token embedding. In the second setting, the factor embedding was equal to the size of the token embedding (1024) and was summed with it.

As shown in Table 5.4, we observe an increase in the string-based evaluation metrics (chrF and BLEU) while COMET scores remain about the same. This is in accordance with Amrhein and Sennrich (2022), who show that COMET models are not sufficiently sensitive to discrepancies in named entities.

Table 5.2 presents the numbers of recognized named entity categories in the training, development and test data.

```
Hlavní|p0 inspektor|p0 organizace|p0 RSPCA|p3 pro|p0 Nový|p2 Jižní|p2 Wales|p2
David|p1 O'Shannessy|p1 televizi|p0 ABC|p5 sdělil|p0 ,|p0 že|p0 dohled|p0 nad|p0
jatky|p0 a|p0 jejich|p0 kontroly|p0 by|p0 měly|p0 být|p0 v|p0 Austrálii|p2
samozřejmostí|p0 .|p0

_Hlavní|p0 _inspektor|p0 _organizace|p0 _R|p3 SP|p3 CA|p3 _pro|p0 _Nový|p2 _Jižní|p2
_Wales|p2 _David|p1 _O|p1 '|p1 S|p1 han|p1 ness|p1 y|p1 _televizi|p0 _A|p5 BC|p5
_sdělil|p0 ,|p0 _že|p0 _dohled|p0 _nad|p0 _ja|p0 tky|p0 _a|p0 _jejich|p0 _kontroly|p0
_by|p0 _měly|p0 _být|p0 _v|p0 _Austrálii|p2 _samozřejmost|p0 í|p0 .|p0
```

Figure 5.1: An example of a sentence tagged with NER source factors before and after subword encoding.

Table 5.2: The number of recognized named entity categories in the training, development and test data. The training data statistics are split into *train-bt*, which was created by noisy back-translation, and *train-parallel*, which is the filtered parallel training data.

	cs				uk			
Category	**train-bt**	**train-parallel**	**dev**	**test**	**train-bt**	**train-parallel**	**dev**	**test**
PER	33,633,602	1,545,658	747	306	30,778,893	1,623,370	827	478
LOC	24,552,404	1,954,319	1,191	454	18,178,736	1,912,604	1,197	771
ORG	29,380,436	1,997,685	566	314	24,117,485	2,221,371	544	606
MISC	-	-	-	-	4,140,394	893,867	168	76
PRO	5,452,326	1,104,860	172	59	-	-	-	-
EVT	1,150,301	111,563	83	10	-	-	-	-

5.3.4 Document-Level Translation

Our work on document-level translation is based on a simple data concatenation method, similar to Junczys-Dowmunt (2019) and Scherrer et al. (2019).

As our training data, we use parallel document-level datasets (GNOME, KDE4, TED2020, QED), as well as synthetically created data, concatenating random sentences to match the desired input length. Specifically, we merge datasets created in the following ways as a single, large dataset:

- Curr → Curr: sentence-level parallel data,

- Prev + Curr → Prev + Curr: previous sentence given as a context,
- 50T → 50T: a fixed window of 50 tokens after subword encoding,
- 100T → 100T: a fixed window of 100 tokens after subword encoding,
- 250T → 250T: a fixed window of 250 tokens after subword encoding,
- 500T → 500T: a fixed window of 500 tokens after subword encoding.

By concatenating such datasets, we allow the model to gradually learn how to translate longer input sequences. It is also capable of sentence-level translation. To separate sentences from each other, we introduced a `<SEP>` tag. An example of a document-level input sequence is shown in Figure 5.2. All data used to train the document-level model were tagged with NER source factors, including the back-translated data.

```
Netvrdím, že bakteriální celulóza jednou nahradí bavlnu, kůži, nebo jiné látky.
<SEP> Ale myslím, že by to mohl být chytrý a udržitelný přírůstek k našim stále
vzácnějším přírodním zdrojům. <SEP> Možná že se nakonec tyto bakterie neuplatní
v módě, ale jinde. <SEP> Zkuste si třeba představit, že si vypěstujeme lampu,
židli, auto, nebo třeba dům. <SEP> Má otázka tedy zní: Co byste si v budoucnu
nejraději vypěstovali vy?
```

Figure 5.2: An example document consisting of five sentences separated with `<SEP>` tags.

5.3.5 Weighted Ensemble

We created a weighted ensemble of four best-performing models. It consisted of the following model types:

- (A) sentence-level models trained with NER source factors (concat 16),
- (B) sentence-level model trained with NER source factors (sum),
- (C) document-level model trained with NER source factors (concat 16).

In this case, the document-level model was used only for the sentence-level translation. The optimal weights for each model were selected using a grid search method. For the specific language pairs, we used the following model and weight combinations:

- Czech → Ukrainian: $1.0 \cdot (2\times\text{A}) + 0.8 \cdot (\text{B}) + 0.6 \cdot (\text{C})$,
- Ukrainian → Czech: $1.0 \cdot (2\times\text{A}) + 0.8 \cdot (\text{B}) + 0.4 \cdot (\text{C})$.

5.3.6 Quality-Aware Decoding

Having the final model ensemble, we created an n-best list containing 200 translations for each sentence with beam search. Then we merged it with a second n-best list containing 50 translations for each sentence, created by a single document-level model with document-level decoding. The idea behind it was that the hypotheses produced by the document-level decoding take into account the context of surrounding sentences, which is not the case with the ensemble. This enabled the use of quality-aware decoding (Fernandes et al., 2022).

We applied a two-stage quality-aware decoding mechanism: pruning hypotheses using a tuned reranker (T-RR) and minimum Bayes risk (MBR) decoding (Kumar & Byrne, 2002, 2004), as shown in Figure 5.3.

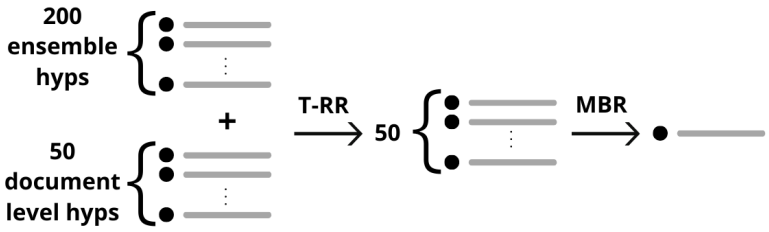


Figure 5.3: A two-stage (T-RR → MBR) quality-aware decoding process. 200 hypotheses generated by the ensemble are merged with 50 hypotheses generated by the document-level model. A tuned reranker is used to prune the total number of hypotheses to 50, and these are then used as input for minimum Bayes risk decoding.

First, we tuned a reranker on the development set, using as features NMT model scores, as well as existing QE models based on TransQuest (Ranasinghe et al., 2020) and COMET (Rei et al., 2020), which are based on Direct Assessment (DA) (Graham et al., 2013) scores or MQM (Lommel et al., 2014) scores. Specifically, we used:

- model ensemble log-likelihood $\log p_\theta(y|x)$ scores,
- TransQuest QE model trained on DA scores (`monotransquest-da-multilingual`),
- COMET QE model trained on MQM scores (`wmt21-comet-qe-mqm`),
- COMET QE model trained on DA scores (`wmt21-comet-qe-da`).

We tuned the feature weights to maximize the COMET reference-based evaluation metric value using MERT (Och, 2003).

After tuning the reranker, we used it to prune the n-best list from 250 to 50 hypotheses per input sentence. The resulting n-best list was used for minimum Bayes risk decoding, using the COMET reference-based metric as the utility function. Minimum Bayes risk decoding seeks, from the set of hypotheses, the hypothesis with the highest expected utility.

$$\hat{y}_{\text{MBR}} = \arg\max_{y \in \bar{\mathcal{Y}}} \underbrace{\mathbb{E}_{Y \sim p_\theta(y|x)}[u(Y, y)]}_{\approx \frac{1}{M} \sum_{j=1}^{M} u(y^{(j)}, y)} \tag{5.1}$$

Equation 5.1 shows that the expectation can be approximated as a Monte Carlo sum using model samples $y^{(1)}, \ldots, y^{(M)} \sim p_\theta(y|x)$. In practice, the translation with the highest expected utility can be chosen by comparing each hypothesis $y \in \bar{\mathcal{Y}}$ with all other hypotheses in the set.

The described two-stage quality-aware decoding process allowed us to further optimize our system for the COMET evaluation metric, which has been shown to have a high correlation with human judgements (Kocmi et al., 2021).

5.3.7 Post-Processing

The final step involved post-processing. We applied the following post-processing steps for each best obtained translation:

- transfer of emojis from the source to the translation using word alignment based on SimAlign (Jalili Sabet et al., 2020),
- restoration of quotation marks appropriate for a given language,

- restoration of capitalization (e.g. if the source sentence was fully uppercased),
- restoration of punctuation, exclamation and question marks (if a source sentence ends with such a mark, we make the translation do likewise),
- replacement of three consecutive dots with an ellipsis,
- restoration of bullet points and enumeration (e.g. if the source sentence starts with a number or a bullet point),
- deletion of consecutively repeated words.

Approach	Sim. score	COMET	chrF
Baseline	-	0.8322	0.5263
Default	0.4	0.8316	0.5260
Best-334	0.19	0.8322	0.5259
Best-133	0.25	0.8323	0.5262

Table 5.3: Results of the on-the-fly adaptation method on the development set. The *default* approach is based on Farajian et al. (2017). However, only 11 sentence pairs were found in this scenario. The experiments denoted as *best-334* and *best-133* used the learning rate values of 0.002 and 10 epochs. In our development set containing 2009 sentence pairs, 334 matching sentences were found in *best-334* and 133 in *best-133*.

5.3.8 On-The-Fly Domain Adaptation

The General MT Task tests the MT system's performance on multiple domains. Therefore, we investigated the possibility of improving our translation system with the on-the-fly domain adaptation method.

This experiment was based on Farajian et al. (2017). Our idea was to retrieve similar sentences from the training data for each input sentence and to fine-tune the model on their translations. After the translation of a single sentence is complete, the model is reset to the original parameters. We used Apache Lucene (McCandless et al., 2010) as our translation memory to search for similar sentences. We indexed all of the training data and used the Marian dynamic adaptation feature. We compared the translation quality with and without the retrieved context. The experiments were carried out with a different similarity score used to choose similar sentence pairs for the fine-tuning process. We empirically modified the learning rate and the number of epochs to find optimal values that improved the translation quality.

Table 5.3 shows the results of the aforementioned experiments on the full development set. We found that only a small number of sentences in the training data were similar to those present in the development set. The results showed that tuning the model on similar sentences from the training data did not significantly improve translation quality. In the end, we decided not to use this method in our WMT 2022 submission.

5.4 Results

The results of our experiments are presented in Table 5.4. We evaluated our models with the COMET[1] (Rei et al., 2020), chrF (Popović, 2015) and BLEU (Papineni et al., 2002) automatic evaluation metrics. ChrF and BLEU scores were computed with the sacreBLEU[2][3] (Post, 2018) tool. We also include scores for the document-level model. In this case, the scores

1: COMET scores were computed with the `wmt20-comet-da` model.

2: BLEU signature: nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.0.0

3: chrF signature: nrefs:1|case:mixed|eff:yes|nc:6|nw:0|space:no|version:2.0.0

include improvements added by back-translation, NER source factors and fine-tuning. The document-level evaluation was split into sentence-level decoding and document-level decoding. In the first scenario, the model translates a single sentence at a time, which is not different from a sentence-level model. In the second scenario, the model translates concatenated chunks of at most 250 subword tokens at a time.

We found that the largest gain in the COMET value was achieved due to the quality-aware decoding method, at the cost of BLEU value. The chrF value remained the same in the Ukrainian→Czech translation direction, while it increased slightly in the Czech→Ukrainian direction. As discussed in section 5.3.3, the inclusion of NER source factors helped the model with the translation of named entities, which is not well reflected in the COMET value, as this metric is not sufficiently sensitive to discrepancies in named entities (Amrhein & Sennrich, 2022).

Table 5.5 shows results for our final submissions compared with the baseline. We performed a statistical significance test with paired bootstrap resampling (Koehn, 2004), running 1000 resampling trials to confirm that our submissions are statistically significant ($p < 0.05$).

Table 5.4: Results of COMET, chrF and BLEU automatic evaluation metrics on the concatenated datasets flores101-dev and flores-101-devtest. ChrF and BLEU metrics were computed with sacreBLEU. Document-level model evaluation includes added back-translation, NER source factors (concat 16) and fine-tuning.

System		uk→cs			cs→uk		
		COMET	**chrF**	**BLEU**	**COMET**	**chrF**	**BLEU**
Baseline (transformer-big)		0.8622	0.5229	24.29	0.7818	0.5175	22.64
+back-translation		0.9053	0.5309	25.41	0.8356	0.5280	23.14
+ner	concat 16	0.9003	0.5314	25.62	0.8362	0.5309	24.28
	sum	0.8991	0.5323	25.87	0.8421	0.5302	23.91
+fine-tune	concat 16	0.9021	0.5344	25.94	0.8387	0.5330	24.51
	sum	0.8990	0.5357	25.99	0.8456	0.5321	24.24
+ensemble		0.9066	0.5376	**26.36**	0.8522	0.5373	**24.85**
+quality-aware		0.9874	0.5376	25.42	0.9238	0.5384	24.50
+post-processing		**0.9883**	**0.5392**	25.89	**0.9240**	**0.5388**	24.63
Document-level	sent-level dec.	0.8942	0.5326	25.47	0.8350	0.5289	23.92
	doc-level dec.	0.8920	0.5324	25.44	0.8356	0.5297	23.78

Table 5.5: Results of COMET, chrF and BLEU automatic evaluation metrics on the test set. ChrF and BLEU metrics were computed with sacreBLEU. The final submission results are statistically significant ($p < 0.05$).

System	uk→cs			cs→uk		
	COMET	**chrF**	**BLEU**	**COMET**	**chrF**	**BLEU**
Baseline (transformer-big)	0.8315	0.5627	31.79	0.8008	0.5849	31.43
Final submission	**1.0488**	**0.6066**	**37.03**	**0.9944**	**0.6153**	**34.74**

5.5 Conclusions

We describe Adam Mickiewicz University's (AMU) submissions to the WMT 2022 General MT Task in the Ukrainian ↔ Czech translation directions. Our experiments cover a range of MT enhancement methods, including transfer learning, back-translation, NER-assisted translation, document-level translation, weighted ensembling, quality-aware decoding, and on-the-fly domain adaptation. We found that using a combination

of these methods on the test set leads to a +0.22 (26.13%) increase in COMET scores in the Ukrainian→Czech translation direction and a +0.19 (24.18%) increase in the Czech→Ukrainian direction, compared with the baseline model. According to the COMET automatic evaluation results, our systems rank first in both translation directions.

References

Aji, A. F., Bogoychev, N., Heafield, K., & Sennrich, R. (2020). In neural machine translation, what does transfer learning transfer? *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7701–7710. https://doi.org/10.18653/v1/2020.acl-main.688 (cited on pages 41, 42)

Amrhein, C., & Sennrich, R. (2022). Identifying weaknesses in machine translation metrics through minimum Bayes risk decoding: A case study for COMET. *arXiv preprint arXiv:2202.05148* (cited on pages 44, 48).

Arkhipov, M., Trofimova, M., Kuratov, Y., & Sorokin, A. (2019). Tuning multilingual transformers for language-specific named entity recognition. *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, 89–93. https://doi.org/10.18653/v1/W19-3712 (cited on page 43)

Aulamo, M., Virpioja, S., & Tiedemann, J. (2020). OpusFilter: A configurable parallel corpus filtering toolbox. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 150–156. https://doi.org/10.18653/v1/2020.acl-demos.20 (cited on page 42)

Edunov, S., Ott, M., Auli, M., & Grangier, D. (2018). Understanding back-translation at scale. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 489–500. https://doi.org/10.18653/v1/D18-1045 (cited on pages 41, 43)

Farajian, M. A., Turchi, M., Negri, M., & Federico, M. (2017). Multi-domain neural machine translation through unsupervised adaptation. *Proceedings of the Second Conference on Machine Translation*, 127–137. https://doi.org/10.18653/v1/W17-4713 (cited on pages 41, 47)

Fernandes, P., Farinhas, A., Rei, R., De Souza, J., Ogayo, P., Neubig, G., & Martins, A. (2022). Quality-aware decoding for neural machine translation. *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1396–1412 (cited on pages 41, 45).

Goyal, N., Gao, C., Chaudhary, V., Chen, P.-J., Wenzek, G., Ju, D., Krishnan, S., Ranzato, M., Guzmán, F., & Fan, A. (2022). The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, *10*, 522–538. https://doi.org/10.1162/tacl_a_00474 (cited on page 42)

Graham, Y., Baldwin, T., Moffat, A., & Zobel, J. (2013). Continuous measurement scales in human evaluation of machine translation. *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, 33–41 (cited on page 46).

Jalili Sabet, M., Dufter, P., Yvon, F., & Schütze, H. (2020). SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. *Findings of the Association for Computational Linguistics: EMNLP 2020*, 1627–1643. https://doi.org/10.18653/v1/2020.findings-emnlp.147 (cited on page 46)

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759* (cited on page 42).

Junczys-Dowmunt, M. (2019). Microsoft translator at WMT 2019: Towards large-scale document-level neural machine translation. *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 225–233. https://doi.org/10.18653/v1/W19-5321 (cited on page 44)

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Aji, A. F., Bogoychev, N., Martins, A. F. T., & Birch, A. (2018). Marian: Fast neural machine translation in C++. *Proceedings of ACL 2018, System Demonstrations*, 116–121. https://doi.org/10.18653/v1/P18-4020 (cited on page 42)

Kocmi, T., Federmann, C., Grundkiewicz, R., Junczys-Dowmunt, M., Matsushita, H., & Menezes, A. (2021). To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. *Proceedings of the Sixth Conference on Machine Translation*, 478–494 (cited on page 46).

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, 388–395 (cited on pages 41, 48).

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., & Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, 177–180 (cited on page 42).

Kudo, T. (2018). Subword regularization: Improving neural network translation models with multiple subword candidates. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 66–75. https://doi.org/10.18653/v1/P18-1007 (cited on page 42)

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71. https://doi.org/10.18653/v1/D18-2012 (cited on page 42)

Kumar, S., & Byrne, W. (2002). Minimum Bayes-risk word alignments of bilingual texts. *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, 140–147. https://doi.org/10.3115/1118693.1118712 (cited on page 45)

Kumar, S., & Byrne, W. (2004). Minimum Bayes-risk decoding for statistical machine translation. *Proceedings of the Human Language Technology Conference of the North American Chapter of the Asso-*

ciation for Computational Linguistics: HLT-NAACL 2004, 169–176 (cited on page 45).

Lommel, A., Uszkoreit, H., & Burchardt, A. (2014). Multidimensional quality metrics (mqm): A framework for declaring and describing translation quality metrics. *Revista Tradumàtica: tecnologies de la traducció*, (12), 455–463 (cited on page 46).

McCandless, M., Hatcher, E., & Gospodnetić, O. (2010). *Lucene in action*. Manning. (Cited on page 47).

Modrzejewski, M., Exel, M., Buschbeck, B., Ha, T.-L., & Waibel, A. (2020). Incorporating external annotation to improve named entity translation in NMT. *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, 45–51 (cited on pages 41, 43).

Och, F. J. (2003). Minimum error rate training in statistical machine translation. *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics – Volume 1*, 160–167. https://doi.org/10.3115/1075096.1075117 (cited on page 46)

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311–318. https://doi.org/10.3115/1073083.1073135 (cited on page 47)

Popović, M. (2015). ChrF: Character n-gram F-score for automatic MT evaluation. *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 392–395. https://doi.org/10.18653/v1/W15-3049 (cited on page 47)

Post, M. (2018). A call for clarity in reporting BLEU scores. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 186–191 (cited on page 47).

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python natural language processing toolkit for many human languages. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 101–108. https://doi.org/10.18653/v1/2020.acl-demos.14 (cited on page 43)

Ramshaw, L., & Marcus, M. (1995). Text chunking using transformation-based learning. *Third Workshop on Very Large Corpora* (cited on page 43).

Ranasinghe, T., Orasan, C., & Mitkov, R. (2020). TransQuest: Translation quality estimation with cross-lingual transformers. *Proceedings of the 28th International Conference on Computational Linguistics*, 5070–5081. https://doi.org/10.18653/v1/2020.coling-main.445 (cited on page 46)

Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2020). COMET: A neural framework for MT evaluation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2685–2702. https://doi.org/10.18653/v1/2020.emnlp-main.213 (cited on pages 41, 46, 47)

Scherrer, Y., Tiedemann, J., & Loáiciga, S. (2019). Analysing concatenation approaches to document-level NMT in two different domains. *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, 51–61. https://doi.org/10.18653/v1/D19-6506 (cited on page 44)

Sennrich, R., & Haddow, B. (2016). Linguistic input features improve neural machine translation. *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, 83–91. https://doi.org/10.18653/v1/W16-2209 (cited on page 43)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc. (Cited on page 42).

Zoph, B., Yuret, D., May, J., & Knight, K. (2016). Transfer learning for low-resource neural machine translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1568–1575. https://doi.org/10.18653/v1/D16-1163 (cited on pages 41, 42)

Development Papers

A Neural Translator Designed to Protect the Eastern Border of the European Union

6

Abstract

This paper reports on a translation engine designed for the needs of the Polish State Border Guard. The engine is a component of the AI Searcher system, whose aim is to search for Internet texts, written in Polish, Russian, Ukrainian or Belarusian, which may lead to criminal acts at the eastern border of the European Union. The system is intended for Polish users, and the translation engine should serve to assist understanding of non-Polish documents. The engine was trained on general-domain texts. The adaptation for the criminal domain consisted in the appropriate translation of criminal terms and proper names, such as forenames, surnames and geographical objects. The translation process needs to take into account the rich inflection found in all of the languages of interest. To this end, a method based on constrained decoding that incorporates an inflected lexicon into a neural translation process was applied in the engine.

6.1 Introduction

The Internet, even in its legal form, may be a source of criminal information. Government bodies all over the world search through Internet sites for potentially criminal texts, to prevent certain acts to which such texts may give rise. For example, the Polish State Border Guard, whose function is to protect the eastern border of the European Union, tracks texts that may concern criminal activities such as general smuggling, trafficking of drugs, medicines, alcohol and cigarettes, people trafficking, human organs trafficking, weapons and explosives, sex crime, document fraud, and trafficking of stolen cars and machines. Two factors make this task difficult for employees of the State Border Guard. Firstly, the texts of interest are sparse and not easy to detect. The problem of the detection of such texts is tackled in Nowakowski and Jassem, 2021a. Secondly, criminal texts may appear in foreign languages, not known to a particular employee. In such cases a machine translation engine may be of significant help to the user.

This paper describes a neural translator designed for the needs of the Polish State Border Guard. The translator is a component of a system designed to search for and store criminal content. The system is being developed within a research project entitled "Advanced Internet analysis supporting the detection of criminal groups"[1] (the project's short name is AI Searcher). The architecture of the AI Searcher system is described in section 6.2. Section 6.3 reports on the data that was used for the training of language pairs applied in the system. Section 6.4 describes how the translation engine was adapted to the domain of criminal texts. Details on the lexicalized translation methods applied in the adaptation are presented in section 6.5. Section 6.6 gives a few examples that show the difference between adapted and unadapted translation. We conclude the paper with some insights relevant to future work.

1: The project is financed by the Polish National Center for Research and Development.

6.2 The AI Searcher Project

The AI Searcher project was launched in December 2018. This three-year program has the aim of developing a system to support the protection of the eastern border of the European Union by searching the Internet for criminal texts that may be of interest to employees of the Polish State Border Guard. The user scenario is the following: The employee of the State Border Guard types an inquiry into an edit window. The Query Expansion Module expands the inquiry to a set of queries that are semantically related to the inquiry. The Translation Module translates the set of queries into Russian, Ukrainian, and Belarusian. The Crawler searches the Internet to find texts in Polish, Russian, Ukrainian, and Belarusian related to the queries. The Translation Module translates the foreign texts back to Polish. Finally, the Classifier analyzes the texts to return those with potentially criminal content.

6.3 Training Data

2: https://opus.nlpl.eu/

3: https://github.com/Helsinki-NLP/Tatoeba-Challenge

The translator engines designed for the system are trained on the OPUS resources.[2] The sets for training, validation and testing are based on the Tatoeba Challenge[3] (Tiedemann 2020). Statistics on the bilingual corpora used in the project are given in Table 6.1.

Table 6.1: Bilingual corpora statistics

Corpus set	Polish–Russian	Polish–Ukrainian	Polish–Belarusian
training set	ca. 19.17m	ca. 1.68m	72,276
validation set	1,000	6,900	287
test set	3,543	2,500	287

The number of sentences for the Polish–Belarusian pair was too low to generate comprehensive translation. A multilingual (Polish–Russian–Ukrainian–Belarusian) model was designed to improve the Polish–Belarusian translation. Its statistics are given in Table 6.2.

Table 6.2: Multilingual corpus statistics

Corpus set	Russian–Belarusian	Russian–Ukrainian	Ukrainian–Belarusian
training set	72,870	ca. 1.52m	66,687
validation set	2,743	6,815	1,000
test set	2,500	10,000	2,355

Table 6.3 shows the BLEU scores of the AI Searcher Translator compared with Google Translate, calculated on the Tatoeba test set.

Table 6.3: Comparison of BLEU scores

Corpus set	pl -> ru	ru -> pl	pl -> uk	uk -> pl	pl -> be	be -> pl
AI Searcher	47.69	43.06	41.25	43.67	24.75	37.92
Google Translate	42.95	43.05	34.84	38.42	35.39	44.19
difference	**+4.74**	+0.01	**+6.41**	**+5.25**	-10.64	-6.27

6.4 Translation of Terminology and Personal Names

The State Border Guard expects that the translation engine should correctly translate proper names, such as surnames, forenames, geographical locations and objects, brands of cigarettes and alcohol, etc. The lists of such names were created semi-automatically: the names underwent automatic transliteration between the Cyrillic and Latin alphabets, and the most frequent names were carefully verified by native speakers. It is worth noting that all verified forenames and surnames were listed and checked together with their inflected forms (there exist 6–7 grammatical cases in all of these languages).

Forenames and surnames in their base Latin form were provided to us by employees of the State Border Guard, names of geographical objects were collected from the available OpenStreetMap resources, and criminal terminology, including brands of cigarettes, cars and alcohol, was gathered from various websites and forums.

Table 6.4 shows the numbers of base forms for verified proper names.

Table 6.4: Statistics of proper names

Proper Names	Polish–Russian	Polish–Ukrainian	Polish–Belarusian
male forenames	1,882	1,902	3,477
male surnames	16,142	29,628	17,421
female forenames	2,117	1,962	3,302
female surnames	19,898	26,114	20,170
geographical objects	5,092	7,613	9,460

The adaptation of the translation engine also took into account a lexicon of criminal terms, which consisted of 1203 entries in each of the language pairs.

6.5 Lexical Constraints

The incorporation of lexicon in neural machine translation has been studied thoroughly in recent years (Arthur et al. 2016, Anderson et al. 2017, Hokamp and Liu 2017, Dinu et al. 2019, Song et al. 2019, Exel et al. 2020). The methodology used in the described experiments was based on a constrained decoding and "code-switching" approach. Our approach was focused on constrained decoding, which uses the Grid Beam Search algorithm introduced by Hokamp and Liu, 2017 and extended by Post and Vilar, 2018 and Hu et al., 2019. We designed an algorithm based on constrained decoding in order to take into account inflected forms of proper names. To evaluate the performance of the algorithm, we carried out experiments in two different scenarios: general and domain-specific. We compared our method with baseline translation, i.e. translation without lexical constraints, in terms of translation speed and translation quality. The lexicalized method resulted in a decrease in translation quality in the general scenario, which shows that augmenting general-domain texts with a specialized lexicon may impair the performance of a neural translator. In the domain-specific scenario the translation

showed significant progress, with an increase of over 3 BLEU points. The cost of the algorithm is a decrease in the translation speed. The details of the experiment are reported in Nowakowski and Jassem, 2021b. There, several manual metrics for the evaluation of terminology translation were introduced: Placement Rate, Duplication Rate and Inflection Rate. The metrics evaluated the proportions of output sentences in which the target lexicon terms were, respectively, properly placed, not duplicated unnecessarily and correctly inflected. The manual evaluation results showed that our method ensures terminological adequacy and consistency as well as linguistic correctness when translating into a morphologically rich language in domain-specific scenarios.

6.6 Examples of Lexicalized Translation

Tables 6.5 and 6.6 show examples of sentences translated with the unadapted and adapted translation engine into Russian and Ukrainian, respectively. The lexicon entries consist of a term in the source language with the equivalent in the target language along with a comma-separated list of its inflectional forms. For each sentence, a manual English translation is given for clarity.

Table 6.5: Examples of lexicalized translation into Russian

Lexicon entry	Georgy -> Георгий, Георгия, Георгию, Георгием, Георгии
Source sentence	**Georgy** Kuzmin przewozi fajki przez wschodnią granicę.
English translation	Georgy Kuzmin transports cigarettes across the eastern border.
Unadapted MT	Джорджи Кузьмин перевозит сигареты через восточную границу.
Adapted MT	Георгий Кузьмин перевозит сигареты через восточную границу.
Lexicon entry	szwarcować -> перебрасывать, перебрасываю, перебрасываешь
Source sentence	Zaczynamy **szwarcować** zioło klientom.
English translation	We are beginning to smuggle the weed to our customers.
Unadapted MT	Мы начинаем портить травы для клиентов.
Adapted MT	Мы начинаем перебрасывать траву клиентам.

Table 6.6: Examples of lexicalized translation into Ukrainian

Lexicon entries	Karpiuk -> Карпюк, Карпюка, Карпюкові, Карпюком hordenina -> горденін горденин гордеїн
Source sentence	Przyniesiemy **hordeninę** do **Karpiuka**.
English translation	We'll bring hordenine to Karpiuk.
Unadapted MT	Ми привеземо гордон до Карпіока.
Adapted MT	Ми принесемо горденін до Карпюка.
Lexicon entry	przećpać -> накачатись, накачатися, накачати, накачаться
Source sentence	Chcesz okazyjnie **przećpać** w promocyjnej cenie?
English translation	Do you want to get high at a discounted price?
Unadapted MT	Ви хочете побути в промоційній ціні?
Adapted MT	Ви хочете накачатися на промоційній ціні?

6.7 Conclusions

In this case study, a translation engine is part of a system that searches for criminal content in Internet documents written in the Polish, Russian,

Ukrainian and Belarusian languages. The adaptation of the translation to the domain of criminal texts consists in the incorporation of lexicon into the neural machine translation engine. The criminal terminology is expected to be translated according to lexical constraints, and the lexical entries should be correctly inflected. An algorithm based on constrained decoding was designed to achieve this goal.

The project described here is ending in December 2021. Work in the near future will focus on further improving Belarusian translation and on increasing efficiency.

References

Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2017). Guided open vocabulary image captioning with constrained beam search. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 936–945. https://doi.org/10.18653/v1/D17-1098 (cited on page 57)

Arthur, P., Neubig, G., & Nakamura, S. (2016). Incorporating discrete translation lexicons into neural machine translation. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1557–1567. https://doi.org/10.18653/v1/D16-1162 (cited on page 57)

Dinu, G., Mathur, P., Federico, M., & Al-Onaizan, Y. (2019). Training neural machine translation to apply terminology constraints. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3063–3068. https://doi.org/10.18653/v1/P19-1294 (cited on page 57)

Exel, M., Buschbeck, B., Brandt, L., & Doneva, S. (2020). Terminology-constrained neural machine translation at SAP. *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, 271–280 (cited on page 57).

Hokamp, C., & Liu, Q. (2017). Lexically constrained decoding for sequence generation using grid beam search. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1535–1546. https://doi.org/10.18653/v1/P17-1141 (cited on page 57)

Hu, J. E., Khayrallah, H., Culkin, R., Xia, P., Chen, T., Post, M., & Van Durme, B. (2019). Improved lexically constrained decoding for translation and monolingual rewriting. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 839–850. https://doi.org/10.18653/v1/N19-1090 (cited on page 57)

Nowakowski, A., & Jassem, K. (2021a). Detection of criminal texts for the Polish state border guard [to appear]. *MIS2-KDD 2021 : The Second International MIS2 Workshop: Misinformation and Misbehavior Mining on the Web* (cited on page 55).

Nowakowski, A., & Jassem, K. (2021b). Neural machine translation with inflected lexicon. *Proceedings of Machine Translation Summit XVIII: Research Track*, 282–292 (cited on page 58).

Post, M., & Vilar, D. (2018). Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1314–1324. https://doi.org/10.18653/v1/N18-1119 (cited on page 57)

Song, K., Zhang, Y., Yu, H., Luo, W., Wang, K., & Zhang, M. (2019). Code-switching for enhancing NMT with pre-specified translation. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 449–459. https://doi.org/10.18653/v1/N19-1044 (cited on page 57)

Tiedemann, J. (2020). The Tatoeba translation challenge – realistic data sets for low resource and multilingual MT. *Proceedings of the Fifth Conference on Machine Translation*, 1174–1182 (cited on page 56).

nEYron: Implementation and Deployment of an MT System for a Large Audit & Consulting Corporation

7

Abstract

This paper reports on the implementation and deployment of an MT system in the Polish branch of EY Global Limited. The system supports standard CAT and MT functionalities such as translation memory fuzzy search, document translation and post-editing, and meets less common, customer-specific expectations. The deployment began in August 2018 with a Proof-of-Concept, and ended with the signing of the Final Version acceptance certificate in October 2021. We present the challenges that were faced during the deployment, particularly in relation to the security check and installation processes in the production environment.

7.1 Business Need

On March 6, 2018, the Polish parliament adopted a law that laid down rules for the Polish Agency of Audit Surveillance regarding the control of auditing companies. The law states that "Documents presented by the audited company for the needs of the surveillance are drawn up in Polish or the audit company provides their translation into Polish." The law forced auditing companies to provide Polish translations for large volumes of English texts. That triggered the idea, at the Polish branch of EY Global Limited (EY Poland), that the cost of the task might be reduced if it were assisted by a translation engine. EY Poland contacted the company Poleng Ltd. (Poleng) to verify the possibility of using their product, TranslAide Workspace, for the task. During initial discussions, EY Poland came to the conclusion that it might be beneficial for the company to have the software installed and running on site.

7.2 The Story of the Deployment

7.2.1 TranslAide Workspace

The first phase of the deployment began in August 2018. The deployed system was based on TranslAide Workspace, which combined computer-aided translation (translation memory with fuzzy search and segment-by-segment editing) with a generic machine translation engine, not trained specifically on the in-domain data. The task consisted in replacing the existing translation engine with a new one, dedicated to the customer.

The deployment was divided into the Proof-of-Concept (POC) and Final Version stages. The POC machine was to be installed in the Linux environment to make the initial deployment easier for the Poleng team. There were no explicit expectations regarding the quality of the translation imposed on the POC version. However, moving forward to the Final Version stage was conditional on acceptance of the POC by the customer – including translation quality, which would be checked by human specialists from the EY corporation. The Final Version – all of the system

components, including model training – was expected to run on the Windows operating system to meet EY's security standards and internal regulations.

The expectations for the system were the following: The TranslAide Workspace system would consist of three modules – Web Application, Translation Memory, and Machine Translation Service:

- Web Application would be the part of the system with which the user interacts;
- Translation Memory would provide translation of segments that were found in its database;
- Machine Translation Service would provide translation of all remaining sentences at a speed not slower than a second per segment.

(Details on current expectations for the three modules are given in section 7.4.)

All system components, as well as the training of the models, should be run on a PC machine with the following specification: NVIDIA GTX 1080Ti GPU, 32 GB RAM and an 8-core processor.

The POC phase ended on schedule (within three months), but the translation quality was not fully satisfactory, as the system sporadically produced incorrect translations of some acronyms and rare words; the issue resulted from certain flaws in subword handling by Marian NMT (Junczys-Dowmunt et al., 2018). On rare occasions, the system would also crash when importing a PowerPoint presentation, because of improper handling of some XML tags specific to the PowerPoint document's internal structure. After the major issues had been identified and fixed, the Final Version was developed for the Windows operating system. It was accepted with a three-month delay in March 2019.

7.2.2 Stand-alone nEYron

Once the POC deployment had been stabilized, the system was given a new name: nEYron. For two years, it was used by several EY employees on a single PC machine that hosted all system components. Meanwhile, nEYron acquired a new look, consistent with the style of other applications dedicated to the same customer. New functional features were developed to satisfy needs arising during the use of the application. An up-to-date list of functionalities is given in section 7.3.

7.2.3 Multi-user Solution

The final phase of deployment took place in 2021. The agreement stated that the application must adhere to EY security standards. The customer expected to receive the following items:

- system installation package;
- system installation instructions;
- system backup policy;
- user's guide;
- disaster recovery procedures.

The creation of the documentation was painless. However, adhering to the security standards was not (see 7.5.2). The process began in April 2021, and the certificate of final acceptance was signed in October 2021.

7.3 System Requirements for the Final Version

7.3.1 EY User Feedback

During the POC stage, EY employees developed a list of requirements that should be added to the system in the Final Version stage. The following three requirements were added after the POC stage: automatic deletion of documents from the user translation history after a specified time (for confidentiality reasons), document sharing between multiple users, and calculation of the approximate cost of translation of a document by a human translator before it is translated by a machine. Cost assessment was intended to help determine to what extent machine translation reduced translation costs over time, compared to human translation. It is based on the number of words included in the document. In addition to the updated list of requirements, EY employees in collaboration with the Poleng team created a mockup of the user interface that would correspond to the look and feel of the other internal EY systems. The user interface was further modified according to the EY guidelines during the development of the Final Version.

7.3.2 Final List of Requirements

The complete and up-to-date list of requirements consists of the following:

- user registration and login, including SSO (single sign-on) login, universal for all services accessible by EY employees;
- document import in `.txt`, `.docx`, `.pptx` and `.xlsx` formats;
- document editing in sentence-by-sentence mode;
- machine translation in an editing window;
- machine translation of entire documents;
- export of the translated document in a format compatible with the imported document;
- pre-translation of documents using translation memory fuzzy search matches;
- ability to proofread and approve translations of sentences;
- expanding translation memory with approved translations;
- transfer of document formatting (fonts, styling, text placement) between input and output document;
- archiving of translated documents per user;
- automatic deletion of documents from user translation history after a specified time;
- document sharing between multiple users;
- calculation of approximate cost of document translation by a human translator.

7.4 System Components

The architecture of the system consists of the following components:

- Machine Translation Service;
- Translation Memory;
- Web Application.

7.4.1 Machine Translation Service

Machine Translation Service provides translations of sentences in the English–Polish and Polish–English directions without human intervention. It is designed as a web service that is invoked by the web application to produce document translations. It is based on the Marian NMT framework (Junczys-Dowmunt et al., 2018). Internally, the web service forwards source sentences from HTTP requests to the Marian websocket server and returns the translations to the web application.

Customer Training Data

In-domain business documents translated by humans were delivered to Poleng in pairs: each document in Polish had its equivalent in English. The document format was either PDF or Microsoft Office (.docx, .doc, .pptx, .xlsx). We applied the following procedure to extract bilingual corpora from business documents:

1: https://tika.apache.org

2: https://github.com/emjotde/eserix

1. Text extraction from business documents using the Apache Tika[1] toolkit.
2. Text segmentation into sentences using eserix[2] – an SRX rule-based sentence segmenter.
3. Text normalization, including punctuation, quoting and commas, using Moses (Koehn et al., 2007) scripts.
4. Alignment of a source text to a target text at the sentence level using the hunalign (Varga et al., 2007) sentence aligner.

This procedure initially allowed us to obtain nearly 70,000 in-domain sentence pairs.

Model Training

Model training consisted of two steps: training of general models on 10 million sentences derived from the OPUS corpora (Tiedemann, 2012), and use of the transfer learning paradigm to fine-tune the general models on the in-domain data. In this way, the system transfers the knowledge from the general model, significantly increasing the translation quality on the in-domain data (such a process has been described, for example, in Aji et al., 2020). As the general model can be reused for future fine-tunings, this technique reduces the total time to solution by a significant margin.

Data preprocessing, in addition to using the Moses (Koehn et al., 2007) normalization scripts, included subword segmentation. We applied subword segmentation to the data using the SentencePiece (Kudo &

Richardson, 2018) tool with the byte-pair encoding (BPE) (Sennrich et al., 2016) algorithm. The vocabulary consisted of 32,000 entries.

All NMT models were trained using the Marian NMT (Junczys-Dowmunt et al., 2018) framework on a single NVIDIA GTX 1080Ti GPU.

For the Proof-of-Concept stage, we trained models based on an RNN-based encoder–decoder architecture with the attention mechanism (Sennrich et al., 2017). We manually assessed translation quality, comparing the model trained only on openly available data with the model fine-tuned on in-domain data as described in section 7.4.1. The annotators evaluated the translations of a test set consisting of 488 sentences, and provided scores for accuracy and fluency by absolute grading on a scale from 0 to 5. The average scores obtained in all of these experiments are presented in Table 7.1. The most significant improvement in the fine-tuned version was achieved for translation accuracy in the Polish–English direction.

Direction	Data	Accuracy	Fluency
PL – EN	Open	3.47	3.61
EN – PL	Open	3.48	3.62
PL – EN	EY	**4.23**	**3.94**
EN – PL	EY	**3.90**	**3.74**

Table 7.1: Results of manual evaluation of preliminary experiments

The results of this manual assessment of the POC version were considered good enough to proceed to the next stage of deployment.

In the final deployment, the NMT model architecture was replaced by the base Transformer (Vaswani et al., 2017), which improved the quality of translation while reducing the time required to train the model. In addition, another 10,000 sentence pairs were derived from new documents provided by the customer. These additional sentences were used for training of the Transformer models.

The results of automatic evaluation based on the BLEU (Papineni et al., 2002) metric, calculated by the SacreBLEU (Post, 2018) tool with default settings, are presented in Table 7.2.

Direction	Data	Architecture	BLEU
PL – EN	Open	RNN	29.72
EN – PL	Open	RNN	26.36
PL – EN	EY	RNN	36.91
EN – PL	EY	RNN	32.99
PL – EN	Open	Transformer	31.13
EN – PL	Open	Transformer	28.34
PL – EN	EY*	Transformer	**39.92**
EN – PL	EY*	Transformer	**35.55**

Table 7.2: Results of automatic evaluation

7.4.2 Translation Memory

Translation Memory is a database of corresponding segments in both languages. The translation of a sentence is added to the memory upon approval by the system user. Search is carried out by an in-house solution: the Anubis system (Jaworski, 2013), which uses a suffix-array-based index for fuzzy matching. Anubis also features a unique algorithm for

the detection and recombination of all sub-segment matches between a candidate sentence and an example from the Translation Memory.

Translation Memory serves two functions in the system: it is used during the translation process, and it also serves as a collection of training data for future fine-tuning of NMT models. During translation of a document, each sentence is first checked in the Translation Memory. If a match is found, the translation is returned as the result and the sentence is not translated by the NMT model.

7.4.3 Web Application

Web Application is the part of the system with which the user interacts. It consists of the following components:

- a server application, following the REST API design, written in the CakePHP framework;
- a user interface, written in the Vue.js framework;
- an SQL database.

All features included in the web application are listed in section 7.3.

Document translation process The main feature of the web application is the document translation process. It consists of the following steps:

1. User imports the document into System;
2. System extracts text from the document;
3. System segments text into sentences using SRX-based rules;
4. System checks the Translation Memory for the existence of each sentence;
5. System sets up batches of sentences whose translations have not been found in the Translation Memory;
6. Batches are sent to the Machine Translation Service;
7. System saves the translations in the database;
8. System prepares the document to be exported at user's request.

Translations found in the Translation Memory and translations produced by the Machine Translation Service are presented to the user in a single window. Once the document has been translated by the machine, the user can post-edit the text segment-by-segment. Each translated segment may be manually approved by the user for it to be stored in the Translation Memory.

Document reconstruction process The system is expected to transfer the document's styling and formatting from the source document to the translated document.

To this end, we make use of the Microsoft Office document structure: the document is unzipped into a set of XML files and the files are iterated in a search for text content. Each found text item is stored in a database and replaced in the XML file with a placeholder tag containing its identifier. When the translation of text items has been completed, the XML files are iterated again, and the placeholder tags are replaced by the translations.

Finally, the XML files are zipped back into the Microsoft Office document package.

7.5 Deployment Challenges

7.5.1 Proof-of-Concept Deployment Challenges

During the POC stage, the entire system was installed on a single PC machine. The initial configuration of the machine and the installation of the system was carried out at Poleng's headquarters in Poznań, Poland. After the system had been installed, the machine was transported to EY's headquarters in Warsaw, Poland. For confidentiality reasons, the machine could not be connected to the Internet and any system updates had to be provided locally. Poleng prepared Docker[3] containers for each of the system components and transported them on a flash drive to the PC machine, when necessary. The use of Docker containers significantly simplified the process, as each deployment of a system update consisted of replacing the Docker container.

3: https://www.docker.com

The only part of the system that could not be updated in this way was the NMT models. For security reasons, training of the model on customer data had to be performed on a PC machine at the EY headquarters. Therefore, the models were not part of the Machine Translation Services container. Instead, they were mounted as a volume in the container so that they could be easily replaced.

7.5.2 Security Check

For the deployment of the multi-user version in the EY infrastructure, each component of the system had to meet a list of security requirements. The necessary modifications to the Translation Memory and Machine Translation Service components were minor, as they involved only changes to the security of the Docker container (the main process running in the container could not run as a root user). The changes to Web Application were more significant, as this component is exposed to the user. The total number of security requirements that the web application had to meet was close to 70. Most of the security requirements (such as the setting of special headers in HTTP responses) were easy to satisfy. However, some security standards proved to be challenging. Among them were:

- replacement of the entire application logging module;
- implementation of the single sign-on (SSO) authentication procedure specific to the EY corporation;
- implementation of database encryption.

A thorough security review was performed by the EY Global technical team after the system had been deployed.

7.5.3 Installation in the Production Environment

Installation of the final version of the system in the production environment included the creation of the installation package and its deployment to the EY infrastructure. The installation package consisted of Docker containers with the system components. Each of the system components was deployed in Docker containers to enable system scalability in the future. The deployment process was executed through screen sharing. Poleng delivered the installation package to the EY technical team and guided them through the installation process.

7.6 Future Plans

Plans for the future include technical improvements to the existing solution, as well as the introduction of new features.

Small improvements may include replacing hunalign (Varga et al., 2007) with vecalign (Thompson & Koehn, 2019) in the bilingual corpus extraction process described in section 7.4.1. We expect that the translation quality of NMT models will improve as a result of better corpus alignment.

To further improve the quality of the NMT models, we intend to use existing monolingual customer documents. We plan to apply the back-translation (Edunov et al., 2018) technique iteratively (Hoang et al., 2018) to increase the quality of our models.

As new terminology emerges, the user expects MT systems to quickly adapt to them. In most cases, data that would cover the new terminology do not yet exist. To solve this problem, we intend to use techniques for forced terminology translation (Bergmanis & Pinnis, 2021; Nowakowski & Jassem, 2021) to ensure that specific terminology is translated according to the needs of the user. Additionally, providing a glossary with specific in-domain terminology would ensure the consistent translation of such terminology when different sentences are translated.

To date, we have relied on the BLEU (Papineni et al., 2002) metric for the evaluation of trained NMT models. To follow current state-of-the-art solutions in MT evaluation, we plan to use the MT Telescope (Rei et al., 2021) to evaluate our models with the COMET (Rei et al., 2020) metric and perform a fine-grained error analysis.

Business documents often have a complex layout structure, whereas current NMT models operate only on sentence-level textual semantics. We want to explore the idea of integrating NMT with Computer Vision to create an end-to-end model which would learn visual features, layout information and textual semantics to produce document-level translations better than the current state-of-the-art methods. Such a model would be able to simplify the process of text extraction, sentence segmentation and document reconstruction, as it would take all document information as an input. To this end, we plan to base our model on the TILT (Powalski et al., 2021) architecture. This was created for the Question Answering task, but we believe that it could be modified for NMT.

7.7 Conclusions

This paper has presented the deployment of an English–Polish translation system at the Polish branch of EY Global Limited. The system supports standard CAT and MT functionalities such as translation memory fuzzy search, document translation and post-editing, and meets less frequent expectations such as single sign-on login and calculation of the cost of human translation for a given document. The paper has presented the challenges that were faced during the deployment, particularly adherence to security expectations and installation in the production environment. Ultimately, the deployment took over three years. Meanwhile, new technologies have been developed in the field of Machine Translation. Once the security issues have been overcome, we hope to be able to update the system with emerging technologies, constantly improving its performance.

References

Aji, A. F., Bogoychev, N., Heafield, K., & Sennrich, R. (2020). In neural machine translation, what does transfer learning transfer? *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7701–7710. https://doi.org/10.18653/v1/2020.acl-main.688 (cited on page 64)

Bergmanis, T., & Pinnis, M. (2021). Facilitating terminology translation with target lemma annotations. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 3105–3111. https://doi.org/10.18653/v1/2021.eacl-main.271 (cited on page 68)

Edunov, S., Ott, M., Auli, M., & Grangier, D. (2018). Understanding back-translation at scale. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 489–500. https://doi.org/10.18653/v1/D18-1045 (cited on page 68)

Hoang, V. C. D., Koehn, P., Haffari, G., & Cohn, T. (2018). Iterative back-translation for neural machine translation. *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, 18–24. https://doi.org/10.18653/v1/W18-2703 (cited on page 68)

Jaworski, R. (2013). Anubis – speeding up computer-aided translation. In *Computational linguistics* (pp. 263–280). Springer. (Cited on page 65).

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Fikri Aji, A., Bogoychev, N., Martins, A. F. T., & Birch, A. (2018). Marian: Fast neural machine translation in C++. *Proceedings of ACL 2018, System Demonstrations*, 116–121 (cited on pages 62, 64, 65).

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., & Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, 177–180 (cited on page 64).

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71. https://doi.org/10.18653/v1/D18-2012 (cited on page 64)

Nowakowski, A., & Jassem, K. (2021). Neural machine translation with inflected lexicon. *Proceedings of Machine Translation Summit XVIII: Research Track*, 282–292 (cited on page 68).

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 311–318. https://doi.org/10.3115/1073083.1073135 (cited on pages 65, 68)

Post, M. (2018). A call for clarity in reporting BLEU scores. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 186–191 (cited on page 65).

Powalski, R., Borchmann, Ł., Jurkiewicz, D., Dwojak, T., Pietruszka, M., & Pałka, G. (2021). Going full-tilt boogie on document understanding with text-image-layout transformer. In J. Lladós, D. Lopresti, & S. Uchida (Eds.), *Document analysis and recognition – icdar 2021* (pp. 732–747). Springer International Publishing. (Cited on page 68).

Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2020). COMET: A neural framework for MT evaluation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2685–2702. https://doi.org/10.18653/v1/2020.emnlp-main.213 (cited on page 68)

Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2021). MT-Telescope: An interactive platform for contrastive evaluation of MT systems. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, 73–80. https://doi.org/10.18653/v1/2021.acl-demo.9 (cited on page 68)

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli Barone, A. V., Mokry, J., & Nădejde, M. (2017). Nematus: A toolkit for neural machine translation. *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, 65–68 (cited on page 65).

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1715–1725. https://doi.org/10.18653/v1/P16-1162 (cited on page 65)

Thompson, B., & Koehn, P. (2019). Vecalign: Improved sentence alignment in linear time and space. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1342–1348. https://doi.org/10.18653/v1/D19-1136 (cited on page 68)

Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, 2214–2218 (cited on page 64).

Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., & Trón, V. (2007). Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series 4*, *292*, 247 (cited on pages 64, 68).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*. Curran Associates, Inc. (Cited on page 65).

POLENG MT: An Adaptive MT Platform

8

Abstract

We introduce POLENG MT, an MT platform that may be used as a cloud web application or as an on-site solution. The platform is capable of providing accurate document translation, including the transfer of document formatting between the input document and the output document. The main feature of the on-site version is dedicated customer adaptation, which consists of training on specialized texts and applying forced terminology translation according to the user's needs.

8.1 General Description

POLENG MT is an MT translation platform available in two versions. Using PaaS (Platform as a Service), the translations are delivered via a cloud web application. In the on-site scenario, the customer organization receives an installation package to be used in the customer's infrastructure. In this case, access to the service is specifically limited to the customer's employees. The following features are shared by both versions of the platform:

- user registration and login;
- document import in `.txt`, `.docx`, `.pptx` and `.xlsx` formats;
- document editing in sentence-by-sentence mode;
- machine translation in an editing window;
- machine translation of entire documents;
- export of the translated document in a format compatible with the imported document;
- pre-translation of documents using translation memory fuzzy search matches;
- ability to proofread and approve translations of sentences;
- expanding translation memory with approved translations;
- transfer of document formatting (fonts, styling, text placement) between input and output document;
- archiving of translated documents per user.

POLENG MT translation models are based on the Marian (Junczys-Dowmunt et al., 2018) and fairseq (Ott et al., 2019) NMT frameworks.

8.2 Customer Adaptation

Adaptation for specific users is carried out in the on-site versions. The task includes the following processes:

- SSO (single sign-on) login integration, if applicable;
- delivery of a translation engine specialized in the customer's domain, fine-tuned on documents provided by the customer;
- incorporation of a customized lexicon into the NMT engine;
- automatic generation of a lexicon from the customer's documents.

The latter two processes take into account the recognition and generation of inflected forms of lexicon entries. This problem is addressed in Nowakowski and Jassem, 2021 and Bergmanis and Pinnis, 2021.

8.3 Supported Languages

Currently, POLENG MT supports the following language pairs, in both directions:

- Polish–English;
- Polish–Ukrainian;
- Polish–Russian.

In the near future, we plan to add support for language pairs with other Eastern European languages, including Czech, Romanian, Bulgarian and Belarusian.

Upon the customer's request, the POLENG MT platform can support any translation direction, on condition that the customer provides suitable parallel data (for example, in the form of business documents and their translations).

References

Bergmanis, T., & Pinnis, M. (2021). Facilitating terminology translation with target lemma annotations. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 3105–3111. https://doi.org/10.18653/v1/2021.eacl-main.271 (cited on page 74)

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Fikri Aji, A., Bogoychev, N., Martins, A. F. T., & Birch, A. (2018). Marian: Fast neural machine translation in C++. *Proceedings of ACL 2018, System Demonstrations*, 116–121 (cited on page 73).

Nowakowski, A., & Jassem, K. (2021). Neural machine translation with inflected lexicon. *Proceedings of Machine Translation Summit XVIII: Research Track*, 282–292 (cited on page 74).

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., & Auli, M. (2019). Fairseq: A fast, extensible toolkit for sequence modeling. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 48–53. https://doi.org/10.18653/v1/N19-4009 (cited on page 73)

Appendices

A

WMT 2022 Certificate

30/11/2022

Dear WMT General MT Task participants,

Artur Nowakowski
Gabriela Pałka
Kamil Guttmann
Mikołaj Pokrywka

On behalf of the organizing committee of the 7th Conference on Machine Translation (WMT22), we would like to thank you for your participation in the WMT22 General Machine Translation Task.

We are pleased to confirm that your submissions to the Czech to/from Ukrainian language pair described in the system description paper:

"*Adam Mickiewicz University at WMT 2022: NER-Assisted and Quality-Aware Neural Machine Translation*", Artur Nowakowski, Gabriela Pałka, Kamil Guttmann and Mikołaj Pokrywka

were ranked at the position 2-3 in the official rankings including human references and unconstrained submissions, and achieved the highest average direct assessment scores among constrained submissions in both language directions. Congratulations on achieving your results.

We look forward to your paper presentation at the 7th Conference on Machine Translation (WMT22) to be held on December 7-8, 2022, in Abu Dhabi, co-located with EMNLP 2022.

Cordially,

Tom Kocmi
on behalf of the WMT22 organizing committee.

B

Declarations of Contribution

Poznań, February 1, 2023

Declaration of Contribution

I hereby declare that the contribution to the following paper:

Artur Nowakowski and Krzysztof Jassem, *Neural Machine Translation with Inflected Lexicon*, Proceedings of Machine Translation Summit XVIII: Research Track, 2021.

is correctly characterized in the table below.

Contributor	Tasks description
Artur Nowakowski	Conceptualization and methodology of the research work, algorithm for inflected lexical constraints incorporation, implementation and experimental setup, conduct of the experiments, human and automatic evaluation analysis, writing of the paper.
Krzysztof Jassem	Research supervision, set-up and conduct of the manual evaluation, writing of the "related work" section of the paper.

Artur Nowakowski

Krzysztof Jassem

Poznań, February 1, 2023

Declaration of Contribution

I hereby declare that the contribution to the following paper:

Artur Nowakowski, *Approaching English-Polish Machine Translation Quality Assessment with Neural-based Methods*, Proceedings of the PolEval 2021 Workshop, 2021.

is correctly characterized in the table below.

Contributor	Tasks description
Artur Nowakowski	Conceptualization and methodology of the research work, conduct of the experiments with Polish-English neural machine translation quality assessment models, automatic evaluation analysis, writing of the paper.

Artur Nowakowski

Artur Nowakowski

Poznań, February 1, 2023

Declaration of Contribution

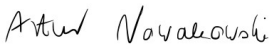
I hereby declare that the contribution to the following paper:

Artur Nowakowski and Tomasz Dwojak, *Adam Mickiewicz University's English-Hausa Submissions to the WMT 2021 News Translation Task*, Proceedings of the Sixth Conference on Machine Translation, 2021.

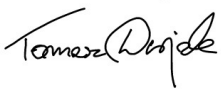
is correctly characterized in the table below.

Contributor	Tasks description
Artur Nowakowski	Conceptualization and methodology of the research work, implementation of the data filtering steps, conduct of the experiments with iterative back-translation, PB-SMT and model ensembling, writing of the paper.
Tomasz Dwojak	Conceptualization and methodology of the research work, conduct of the experiments with transfer learning, preparation of the baseline NMT models, writing of the paper.

Artur Nowakowski



Tomasz Dwojak



Poznań, February 1, 2023

Declaration of Contribution

I hereby declare that the contribution to the following paper:

Artur Nowakowski, Gabriela Pałka, Kamil Guttmann and Mikołaj Pokrywka, *Adam Mickiewicz University at WMT 2022: NER-Assisted and Quality-Aware Neural Machine Translation*, Proceedings of the Seventh Conference on Machine Translation, 2022.

is correctly characterized in the table below (* and † denote groups of equal contribution).

Contributor	Tasks description
Artur Nowakowski*	Conceptualization and methodology of the research work, idea behind the system as a whole, integration of the separate components into a single system, implementation of the data filtering process, conduct of the experiments with transfer learning, back-translation, quality-aware decoding and model ensembling, writing of the paper.
Gabriela Pałka*	Conceptualization and methodology of the research work, implementation of the NER processing module, conduct of the experiments with NER-assisted translation, integration of NER annotations as source factors into the model architecture, writing of the paper.
Kamil Guttmann†	Conduct of the experiments with document-level translation, implementation of post-processing steps.
Mikołaj Pokrywka†	Conduct of the experiments with on-the-fly domain adaptation, optimization of the data filtering process, optimization of quality-aware decoding hyperparameters.

Gabriela Pałka

Artur Nowakowski

Kamil Guttmann

Mikołaj Pokrywka

Poznań, February 1, 2023

Declaration of Contribution

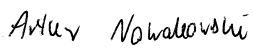
I hereby declare that the contribution to the following paper:

Artur Nowakowski and Krzysztof Jassem, *A Neural Translator Designed to Protect the Eastern Border of the European Union*, Proceedings of Machine Translation Summit XVIII: Users and Providers Track, 2021.

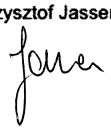
is correctly characterized in the table below.

Contributor	Tasks description
Artur Nowakowski	Conceptualization and methodology of the research work, implementation of the MT system, conduct of the experiments, integration of lexicons into the MT system, analysis of the automatic evaluation results, writing of the paper.
Krzysztof Jassem	Supervision of the R&D work, set-up up and supervision of the lexicon acquisition process, writing of the paper.

Artur Nowakowski



Krzysztof Jassem



Poznań, February 1, 2023

Declaration of Contribution

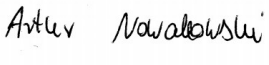
I hereby declare that the contribution to the following paper:

Artur Nowakowski, Krzysztof Jassem, Maciej Lison, Rafał Jaworski, Tomasz Dwojak, Karolina Wiater and Olga Posesor, *nEYron: Implementation and Deployment of an MT System for a Large Audit & Consulting Corporation*, Proceedings of the 23rd Annual Conference of the European Association for Machine Translation, 2022.

is correctly characterized in the table below.

Contributor	Tasks description
Artur Nowakowski	Provision of the MT capabilities of the system, machine translation models training, development of the machine translation service, development of the front-end application, deployment of the final version of the system, writing of the paper.
Krzysztof Jassem	Analysis and design of the system requirements, supervision of the work, writing of the paper.
Maciej Lison	Analysis and design of the system requirements, design of the system architecture, design and implementation of the web application, integration of the system components.
Rafał Jaworski	Implementation of the translation memory component, set-up of the manual evaluation, initial system deployment.
Tomasz Dwojak	Initial training of machine translation models, initial system deployment.
Karolina Wiater	Application testing, user interface design, provision of the customer feedback, manual evaluation of the system.
Olga Posesor	Application testing, user interface design, provision of the customer feedback, manual evaluation of the system.

Artur Nowakowski



Krzysztof Jassem

Maciej Lison

Rafał Jaworski

Tomasz Dwojak

Karolina Wiater

Olga Posesor

Poznań, February 1, 2023

Declaration of Contribution

I hereby declare that the contribution to the following paper:

Artur Nowakowski, Krzysztof Jassem, Maciej Lison, Kamil Guttmann and Mikołaj Pokrywka, *POLENG MT: An Adaptive MT Platform*, Proceedings of the 23rd Annual Conference of the European Association for Machine Translation, 2022.

is correctly characterized in the table below.

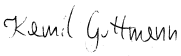
Contributor	Tasks description
Artur Nowakowski	Training of the MT models, development of the machine translation service, development of the front-end application, deployment of the final version of the system, writing of the paper.
Krzysztof Jassem	Analysis and design of the system requirements, work supervision, writing of the paper.
Maciej Lison	Analysis and design of the system requirements, design of the system architecture, design and implementation of the web application, integration of the system components.
Kamil Guttmann	Application testing, implementation of lexicon management functionality.
Mikołaj Pokrywka	Application testing, implementation of lexicon management functionality.

Artur Nowakowski

Krzysztof Jassem

Maciej Lison

Kamil Guttmann



Mikołaj Pokrywka