Domain Adaptation for Machine Translation Involving a Low-Resource Language: Google AutoML vs. from-scratch NMT Systems

Margita Šoštarić
Cerence Inc.
Aachen, Germany
gita.sostaric@gmail.com

Nataša Pavlović
University of Zagreb
Zagreb, Croatia
npavlovi@ffzg.hr

Filip Boltužić
University of Zagreb
Zagreb, Croatia
filip.boltuzic@fer.hr

Abstract

Despite the advances in machine translation (MT) made with neural models, adaptation of such systems for specialist domains is challenging. The problem is heightened for low-resource languages. Additionally, the computational resources and expertise needed to train neural models present barriers for smaller translation companies and freelancers, for whom paid but affordable customization services might present a viable solution. One such service, Google Cloud AutoML, is here compared to domain adaptation of neural MT systems trained from scratch using OpenNMT, an open-source MT toolkit. The from-scratch systems are trained on a larger out-of-domain and a smaller in-domain dataset comprised of medical texts. The same in-domain data are used to customize Google Translate. System performance is compared using automatic and human evaluation. The resources, skills, costs and time necessary to set up the examined systems are discussed.

7. Introduction

Despite the considerable advances in machine translation (MT) achieved with neural models, adaptation of such systems for specialist domains remains a challenge (Luong and Manning, 2015; Koehn and Knowles, 2017; Chu *et al.*, 2017). This is particularly true of low-resource languages, which lack high-quality training data both out of and in domain. The computational resources, expertise and time needed to train neural machine translation (NMT) models may present barriers to entry for smaller translation companies and freelancers. For them, paid but relatively low-cost services, offering pre-trained systems that can be used off the shelf or customized with no programming skills, might present a viable solution.

This study aims to test the performance of such systems, offered by Google, and compare them to own NMT systems built from scratch. The systems are trained for the low-resource English-Croatian (EN-HR) language pair and adapted for the medical domain using a small in-domain dataset. While we have no insight into Google's training methods, we tried out several well-known domain adaptation methods to train our models.

The performance of the general and domain-adapted systems obtained by the two approaches is tested using automatic and human evaluation. The results are discussed with regard to the resources, skills, costs and time needed for setting up either type of system.

8. Previous research on domain adaptation

With the growing demand for high-quality domain-adapted MT (Koehn and Knowles, 2017), this topic has attracted significant attention among researchers. Various approaches have been proposed, aimed at adapting the data, the model architecture or the translation process to attain better performance on domain-specific texts (see Chu and Wang, 2018 for a more comprehensive overview of recent methods).

Among the most popular approaches is fine-tuning (Luong and Manning, 2015; Sennrich *et al.*, 2016a; Servan *et al.*, 2016; Freitag and Al-Onaizan, 2016). It involves training a general-purpose model on out-of-domain parallel data, usually available in larger quantities, and then tuning it on a smaller in-domain corpus. However, fine-tuning on a small dataset quickly leads to overfitting, meaning that the system performance drastically deteriorates on data that differ from the in-domain training dataset. To avoid this, some approaches use a combination of the in-domain and out-of-domain data, where the ratio (Chu *et al.*, 2017) or the weight of in-domain data (Wang *et al.*, 2017b) is increased in training. This way the system "attends" more to the desired data, but their effect is moderated. Following Jean *et al.* (2015), another approach is to combine the general-purpose and the fine-tuned model in translation (Freitag and Al-Onaizan, 2016).

Domain adaptation is frequently framed as a multi-domain problem, where systems are tuned to perform better for different domains. Here tags can be appended at sentence or word level (Kobus *et al.*, 2017) to give the system additional meta-information, prompting it to produce translations with domain-appropriate vocabulary and style. There have also been attempts to identify sentences in the out-of-domain data that are similar to the in-domain. This involves assigning scores to sentences based on similarity of sentence embeddings (Wang *et al.*, 2017a), that is, the model's internal representations of the data, or on sentence perplexity (Axelrod *et al.*, 2011), that is, the estimation of complexity of an out-of-domain sentence according to the language model trained on the in-domain data.

In-domain data can also be boosted by creating synthetic parallel corpora (Sennrich *et al.*, 2016a). This is done by automatically back-translating the monolingual in-domain data available in the target language. As using data with poor source-side quality does not seem to hurt MT performance, this is a viable option where parallel data are scarce. Recently, Imankulova *et al.* (2019) proposed addressing both problems by leveraging out-of-domain data and a pivoting language to train systems that are then fine-tuned to produce domain-adapted translations in a low-resource setting. Finally, as terminology plays a prominent role in domain adaptation, some approaches utilize bilingual terminological glossaries. They involve word replacements (Arčan *et al.*, 2017, Hashimoto *et al.*, 2016), or can be extended to identifying and fixating the translations of larger pieces of text retrieved from translation memories (Silva, 2019).

Despite abundant research on domain adaptation, few studies provide human feedback on the proposed methods. Two exceptions are Etchegoyhen *et al.* (2018) and Castilho *et al.* (2019), who perform an extensive human evaluation of MT systems adapted to different domains for various languages. Both studies involve direct ranking and scale-based subjective evaluations of translations on different criteria (e.g. adequacy and fluency) to estimate system quality.

9. Datasets

To simulate a realistic situation at translation companies, we used the data they are likely to have at their disposal. The in-domain dataset can be constructed from in-house translation memories and term bases for the domains the company specializes in. The out-of-domain data can be freely obtained from the websites listed in subsection 3.2 for various language pairs.

A. In-domain dataset

The medical in-domain dataset was obtained from Cochrane, an organization promoting evidence-based medicine. Cochrane produces, among other, plain language summaries of medical reviews and translates them from English into 14 languages, including Croatian (Cochrane Collaboration, 2019). The translations are done by volunteers and reviewed by medical experts, which contributes to their overall quality.

The EN-HR dataset received in March 2018 was cleaned of formatting tags, as well as resegmented and realigned. We used LF Aligner[1], manually checking and fixing the alignment where required. The resulting dataset consisted of 46K unique high-quality segment pairs[2].

Additionally, we cleaned and formatted several medical glossaries, compiled by students as part of their coursework and checked by medical experts. In total, we had 12K English medical terms with their Croatian equivalents.

B. Out-of-domain datasets

Given that NMT is highly dependent on large quantities of training data (Koehn and Knowels, 2017), our in-domain corpus was not sufficient to train systems from scratch. Although domain adaptation benefits from out-of-domain data that is similar to the in-domain data, this requirement is not easily met in low-resource scenarios. We hence used almost all EN-HR parallel resources available at the time of research on Opus corpus[3], ELRC-SHARE[4] and CLARIN repository[5], as well as the JRC-Acquis[6] dataset. The only resource that was not used entirely was the OpenSubtitles2016 corpus, from which only a subset (150K of the 7M sentence pairs) was selected in order to mitigate its predominance over the other corpora, as the style and structure of spoken language might be less useful for the medical domain.

C. Data pre-processing

We applied some of the standard data preparation methods for MT: we tokenized and truecased the data, as well as replaced some special characters (e.g. different quotation marks) with uniform codes.[7] We further removed the lines that did not contain any words and fixed the encoding errors noticed when manually checking the data. Next, we filtered out all segments with less than three and more than 80 tokens.[8] The same pre-processing steps were applied to the in-domain and out-of-domain data. Subsequently, we had a total of 1,715,169 unique segment pairs of out-of-domain and 45,674 unique segment pairs of in-domain data.

We divided these data into three separate parts: the train set, used for training the system, the validation set, used for checking the system performance during training, and the test set, used to evaluate the performance of the final MT system. The sets were created separately for the in-domain and out-of-domain data. The total number of segment pairs in each set can be seen in Table 1. It should be noted that AutoML creates these sets automatically (see 4.2), so the in-domain test set was extracted from the test set AutoML had created, to ensure that all systems can be tested with the same data.

[1] https://github.com/kindlychung/af-aligner/blob/master/LF_aligner_readme.txt

[2] In this paper the terms "segment" and "sentence" are used interchangeably, as are "token" and "word".

[3] Eubookshop, DGT, GNOME, hrenWaC, KDE4, SETIMES, Tatoeba, OpenSubtitles2016, Ubuntu; http://opus.nlpl.eu/

[4] Acts on Biological and Landscape Diversity and Environmental Protection, Nature protection strategy of Croatia, the websites of the Croatian Bureau of Statistics, Croatian Institute of Public Finance, Croatian Journal of Fisheries, Croatian Mine Action, Croatian Ministry of Public Administration, Croatian Ministry of Public Administration, Croatian National Bank, Croatian Rural Development Programme, Embassy of Finland Zagreb, Government Office for Cooperation with NGOs, Journal of the Croatian Association of Civil Engineers, Ministry of Foreign and European Affairs, National and University Library in Zagreb, Swedish Crime Victim Compensation and Support Authority; https://elrc-share.eu/

[5] Tourism English-Croatian Parallel Corpus; https://www.clarin.si/repository/xmlui/handle/11356/1049

[6] https://ec.europa.eu/jrc/en/language-technologies/jrc-acquis

[7] For these steps we used the scripts provided in the toolkit for statistical MT MOSES (Koehn *et al.*, 2007).

[8] To keep the average segment length closer to that of the in-domain data (ca. 22 tokens), the OpenSubtitles2016 corpus was filtered more strictly, with the lower bound set to seven tokens.

	Train	Validation	Test
Out-of-domain data	1,707,169	4,000	4,000
In-domain data	42,674	1,000	2,000

Table 1. The sizes of the train, validation and test sets for in-domain and out-of-domain data.

Finally, the data were segmented into sub-word units with byte-pair encoding (Sennrich *et al.*, 2016b), as this method helps circumvent the limited vocabulary problem. The BPE algorithm was used with 80K merging operations and learned separately for the source and target side.

10. System training

A. From-scratch models

To train our NMT models we used the Transformer architecture (Vaswani *et al.*, 2017), which has achieved state-of-the-art status since its introduction, outperforming architectures based on recurrent or convolutional neural networks on different tasks (Hieber *et al.*, 2017; Chen *et al.*, 2018). We used the implementation provided in the OpenNMT toolkit (Klein *et al.*, 2018) because the framework is intuitive and offers a list of training parameters that enable the user to replicate the results from the original Transformer paper.[9] This in turn means that substantial experience in training NMT systems is not necessary to use the toolkit, which makes it a viable solution for translation companies.

1. Transformer architecture

Although it has the traditional encoder-decoder architecture, the Transformer relies primarily on attention layers to translate source into target sentences. To learn the representations of source sentences, it uses stacked encoders, each consisting of two sublayers: a self-attention layer and a feed forward neural network. In the self-attention layer, the representation of each word is learned by considering the other relevant words in the sentence. The decoding side has a similar structure, with the inclusion of another encoder-decoder attention sublayer after the self-attention layer, enabling the decoder to attend to specific elements in the source sentence encoding during translation. This mechanism has already been recognized as one of the essential components of MT models based on recurrent neural networks (Bahdanau *et al.*, 2015; Luong *et al.*, 2015). An important difference in the decoder's self-attention layer is that the input in training is masked, i.e. for every word, the decoder can only look at all the previously produced words and the source representations, so it cannot see the words that come after the position it is currently processing. Its structure makes the Transformer an efficient and powerful architecture because the processing of each word is independent, allowing for parallelization, and because the relations between each of the words are modelled directly, facilitating the capturing of features inherent in the local context. For a more detailed description of the architecture, we refer the reader to the original paper by Vaswani *et al.* (2017).

[9] Although the effect of these parameters varies for different data, they are a good starting point for experimenting with the framework. We used six encoding and decoding layers and word vector size of 512; we trained for 200,000 steps and saved the model every 10,000 steps. The full list of parameters can be found here: http://opennmt.net/OpenNMT-py/FAQ.html#how-do-i-use-the-transformer-model-do-you-support-multi-gpu. We used two GPUs, and the training lasted a day and a half.

2. Domain adaptation methods

We first trained a general-purpose model using all training data. We chose not to train a model without any in-domain data because the training data for Google Translate's generic system likely also included medical texts, and because we wanted to explore if this straightforward training procedure would suffice to produce good domain-specific translations.

We then conducted multiple experiments by applying several simple methods, keeping the training parameters and vocabulary size (80K) constant for all systems. The model that yielded the best results on the in-domain test set according to automatic metrics was partly based on the approach by Chu *et al.* (2017). They used a combination of in-domain and out-of-domain training data, but increased the ratio in which the system learned from in-domain data by oversampling. This way the system still trains on a large quantity of data, but effectively tends to the in-domain data more than if it only constituted a small portion of the overall training set. Unlike Chu *et al.* (2017) we did not use domain tags, but we utilized the data from medical glossaries to identify sentence pairs containing English terms on the source side and their Croatian equivalents on the target side[10], adding them to the in-domain data in oversampling. While this simple method hardly compares to the more sophisticated ways of estimating similarity with the in-domain corpus, it enabled us to increase the frequency of medical terms used in context in the training data. The total ratio between the out-of-domain and oversampled data in training was roughly 50-50.

The other experiments involved domain tagging similar to Kobus *et al.* (2017), and OpenNMT's dynamic dictionary functionality[11], but neither method outperformed the oversampling one. Several methods, including fine-tuning, were also tried with recurrent architectures, but their performance (checked manually and automatically) was far below the Transformer (TF) models.

To keep the evaluation setup of our from-scratch systems comparable to the Google-based systems described below, and to make human evaluation feasible, we focus only on the general-purpose model (TF_{gen}), and the best performing domain-adapted system (TF_{med}), trained with oversampled data.

B. Google-based models

Two Google-based NMT models were used: the generic model of the well-known translation service Google Translate (GT_{gen}) and its customized version (GT_{med}), adapted for the medical domain using Google Cloud AutoML. This "suite of machine learning products that enables developers with limited machine learning expertise to train high-quality models" (Google Cloud, 2019b) offers a range of products, including MT. The service is commercial but affordable at $76.00 per hour of training (Google Cloud, 2019a). A 12-month free trial with $300 credit is also available, making the initial model customization effectively free of charge (Google Cloud, 2019c).

We simply uploaded the medical dataset to the platform, and the domain adaptation of the generic GT system was launched. The AutoML service automatically creates the training (80%), validation (10%) and test sets (10%). Automatic evaluation of the resulting customized model shows an improvement of +3.96 BLEU points on the test set in comparison to the generic GT model.[12] The training of the customized system took about three hours. No special knowledge, or indeed any further involvement on our part was required to obtain the system.

[10] For this we first lemmatized the Croatian data using UDPipe's lemmatizer (Straka and Straková, 2017).

[11] https://github.com/OpenNMT/OpenNMT-py/issues/265

[12] We do not focus on AutoML's scores because we performed a separate automatic evaluation (see 6.1) to ensure comparability.

An API key enabling MT integration in the CAT tools that support this functionality is available for both the customized and the generic GT model. The first one million characters are free, whereas a charge of $80 and $20 per next one million characters applies to the customized and generic GT systems respectively (Google Cloud, 2019a).

11. Evaluation

We automatically evaluated the four systems, $\mathrm{GT}_{\mathrm{gen}}$, $\mathrm{GT}_{\mathrm{med}}$, $\mathrm{TF}_{\mathrm{gen}}$ and $\mathrm{TF}_{\mathrm{med}}$, both on in-domain and out-of-domain test sets to verify that the applied methods for domain adaptation did not harm the systems' overall performance. We used three automatic metrics, BLEU (Papineni *et al.*, 2002), chrF (Popović, 2015) and TER (Snover *et al.*, 2006). As the most frequently used metric in MT evaluation, BLEU is a good rough estimate of system quality. Unlike BLEU, which compares sequences of tokens, chrF operates at character level, which makes it more flexible in handling morphologically rich languages.[13] As an edit rate, TER estimates the post-editing effort. The lower the TER score the better, as it represents the changes that would have to be made to the MT output. All metrics are language independent and freely available. It should be noted that most pre-processing steps were removed for evaluation: BPE and tokenization were reverted, and the original casing and special characters restored. This output format is more useful to translators, and it is consistent with the output of GT systems, ensuring all systems are evaluated on equal terms.

As we primarily wanted to examine system performance on domain-specific data, the medical test set was additionally assessed by human evaluators. A survey consisting of two parts was created using LimeSurvey[14]. The first, shorter part compiled data on the respondents' background and experience in medical translation. In total, 27 evaluators took part in the study: 12 translation students, 7 professional translators, 5 medical professionals, 1 student of medicine, 1 student of pharmacy and 1 "other". Three medical professionals, as well as both non-translation students and the respondent of "other" background had volunteered as Cochrane translators. The remaining two medical professionals had had considerable experience with medical translation unrelated to Cochrane. All the translation students had translated Cochrane texts during their coursework at the University of Zagreb. All the professional translators had had experience translating medical texts.

The second part contained translation evaluation tasks, each consisting of an English source sentence and two of its machine translations. The evaluators were asked to decide which of the translations was better according to two criteria: accuracy and usability. We modified the usual accuracy-fluency dichotomy because MT systems today are of reasonable quality and generally produce fluent output. We concluded that comparing translations of similar fluency would not yield relevant results in terms of translation quality, while also posing a demanding task for the evaluators. We believe that the proposed criteria of accuracy and usability make sense in the context of two major use cases of MT. In practice, MT is increasingly used in its raw form to extract basic information from a text in an unknown language, in which case accuracy is of the utmost importance, especially in the healthcare domain. The other major use of MT is to facilitate the translation workflow, in which case the crucial factors are the effort and speed with which translation errors can be corrected. Hence, as explained to the evaluators in the instructions, a translation is more *usable* if the errors it contains could be edited more quickly and easily to bring it to publishable quality. On the other hand, a translation is more *accurate* if it preserves the source information in a more exact and complete way than the other translation. With this distinction, a

[13] We used chrF3, as it was found to correlate well with human judgement for morphologically rich languages (Stanojević *et al.*, 2015).

[14] https://www.limesurvey.org/

less accurate translation can be more usable if the error that makes it inaccurate is easy to correct (for instance, if a negation is missing or if a term is mistranslated).

For each of the two criteria, the evaluators were offered three options: 1) the first translation is better; 2) the second translation is better; 3) both translations are equally good. The respondents were instructed to avoid overusing the last option. To prevent attrition, the survey could be interrupted and resumed later. The process took between 90 and 120 minutes. Reference human translations were not given, but print and electronic resources could be consulted if needed.

Pairwise comparison was chosen over direct ranking of all four translations to make the evaluation process easier for the respondents (Pighin *et al.*, 2012; Šoštarić, 2018). The questions were given in randomized order to reduce bias, but all four translations of the source sentence were ultimately compared by the same evaluator. The complete ranking was determined from the averaged scores of Elo rating[15] that was calculated for each group of questions relating to the same source sentence.

In order to obtain a representative number of evaluations, customized surveys were created for each evaluator. They consisted of the shared and individual subset, both of which contained ten[16] source sentences and their respective translations. All surveys had to have a similar average source sentence length, and sentences for which two or more systems produced identical translations were avoided. In total, we obtained evaluations for the translations of 285 sentences. The ten shared sentences were used to calculate the inter-rater agreement using Fleiss' kappa.

12. Results and discussion

A. Automatic evaluation

The results of automatic evaluation are given in Table 2. For the out-of-domain test set, it is interesting that the TF systems perform on par with GT_{gen}, which was undoubtedly trained on much larger quantities of data. In comparison, the performance of GT_{med} is much poorer according to the metrics. This might suggest that AutoML's domain-adaptation methods make the system better attuned to the in-domain data, but at the expense of overall performance. In contrast, training a TF system on a combination of out-of-domain and oversampled in-domain data did not harm its performance on the out-of-domain test set. For the in-domain dataset, we can see that both domain-adapted systems outperform their generic variants, and that GT_{med} seems to perform similarly to TF_{gen}.

	Out of domain			In domain		
	BLEU	chrF3	TER	BLEU	chrF3	TER
GT_{gen}	28.63*	55.40	0.604*	22.40*	52.08*	0.619*
GT_{med}	22.13*	49.98*	0.673*	24.00	53.86	0.604
TF_{gen}	33.17	55.26	0.568	25.08	53.83	0.592
TF_{med}	**33.60**	**55.96**	**0.565**	**27.96***	**55.58***	**0.577***

Table 2. The results of automatic evaluation. The best results are bolded.
Statistical significance at $p < 0.05$ was measured between all systems. Only the scores that are statistically significant in comparison to all other systems are marked with asterisks.

[15] We used the following implementation: https://github.com/sublee/elo. This rating method is used in sports.

[16] In the case of one evaluator, 15.

Overall, the performance of from-scratch systems is closer to that of Google-based systems than anticipated, with both TF systems scoring higher than GT ones in almost all cases. However, the differences are not always statistically significant[17] and the scores should be interpreted with some caution, given that we have no insight into what the GT systems were trained on and how. For example, the BLEU scores of our Transformer systems are much higher on the out-of-domain data, but this could only point to the translations being less varied, and not necessarily better. Nevertheless, the results certainly indicate that training an own system from scratch constitutes at least as good an option as using services such as AutoML in scenarios where domain-adapted MT is needed.

B. Human evaluation

In comparison to automatic evaluation, according to which TF_{med} performed significantly better than the other systems, GT_{med} and TF_{gen} performed similarly and GT_{gen} performed worst on the in-domain dataset, human evaluation ranks GT_{gen} highest, followed by TF_{med} and GT_{med}, with TF_{gen} bringing up the rear on both of the evaluated criteria (Table 3). These discrepancies highlight the well-known concerns regarding the ability of automatic metrics to reflect human judgement of translation quality.

	Accuracy		Usability	
	Elo rating	Rank	Elo rating	Rank
GT_{gen}	1201.65	1	1202.09	1
GT_{med}	1200.08	3	1199.68	3
TF_{gen}	1197.91	4	1198.19	4
TF_{med}	1200.34	2	1200.02	2

Table 3. Overall Elo ratings and system ranking calculated on all survey questions. The numbers used for rating are arbitrary and should only be interpreted in relative comparison to each other.

However, the differences between ratings are small, and the inter-rater agreement is only "fair" for both accuracy (0.301) and usability (0.237), making the results of human evaluation inconclusive. Nevertheless, we believe it was a valid decision to evaluate system output on the two criteria, as they sometimes highlighted more fine-grained distinctions between the translations. An example is shown in Table 4: as GT_{gen} translated "labour" as "work" instead of "giving birth", most evaluators judged this translation as the least accurate. However, it was better structured overall than the translations produced by the other systems, which made it the most usable.

[17] On the out-of-domain dataset there is no statistical significance between the two TF systems according to the metrics, nor do they significantly differ from GT_{gen} according to chrF3. On the in-domain dataset GT_{med} and TF_{gen} are not significantly different.

Source sentence	There is little doubt that women should be encouraged to utilise positions which give them the greatest comfort, control and benefit during first stage **labour**.
GT_{gen} translation	Nema sumnje da žene treba poticati da koriste položaje koji im pružaju najveću udobnost, kontrolu i dobrobit tijekom prve faze **rada**. [There is no doubt that women should be encouraged to utilise positions which give them the greatest comfort, control, and benefit during the first stage of **work**.]

Table 4. Example of the GT_{gen} translation judged as the least accurate, but the most usable translation option.

The discrepancy between the results of automatic and human evaluation, and the variability in human responses led us to investigate the data further. Manual examination of the translations revealed that the performance of all four systems varied both within the produced translations (e.g. a system translated one part of the sentence better, but mistranslated the other, while another system did the opposite) and between the translations (e.g. systems produced perfect translations of some sentences, but omitted significant amounts of information in others). Consequently, the standard deviation is very high for all automatic metrics, especially for the less flexible BLEU.[18] It is hence not surprising that the human evaluations varied as well, as they partly depended on the data assigned to each participant. As a smaller dataset might have failed to reflect this, we believe it was a good decision to aim for larger coverage of the test set by creating customized surveys.

Finally, we note that nine of the 27 evaluators ranked both GT systems above the TF systems on both accuracy and usability. However, looking only at the 10 common questions, the ranking for both criteria is TF_{med}-GT_{gen}-GT_{med}-TF_{gen}. We therefore conclude that the performance of all four systems is indeed similar on the in-domain data, and that any of the options presented here would provide a good starting point for smaller translation companies to explore the use of MT.

13. Conclusion

In this study we explored the options available to smaller translation companies and freelancers needing high-quality domain-adapted MT. We compared two systems obtained from an online service to two models trained from scratch using a state-of-the-art NMT architecture. Google's AutoML service only requires basic computer literacy to train the models, which can be integrated in the translation workflow using an API key. Although affordable, the service is still paid, which might hinder frequent model updating. In that respect it is easier to train own systems, especially as this enables a better control over the data and methods used in training. There are numerous NMT toolkits available, but we opted for OpenNMT because it is intuitive, fast and implements a range of NMT architectures and features. Nevertheless, the training and data pre- and post-processing take longer than with AutoML, and require basic programming skills and special resources such as access to a GPU.

We also examined viable domain-adaptation methods using a small medical dataset. With AutoML, domain adaptation only requires the in-domain dataset to be uploaded and the platform takes care of everything else. For the from-scratch systems we applied several simple methods, using both in-domain and out-of-domain data for training. The best results were obtained by oversampling the in-domain data, which biased the system to pay equal attention to the medical

[18] Standard deviation for all four systems is above 20 points. The deviation for chrF3 and TER is less pronounced (around 16 and 0.35 points, respectively), but still relatively high.

and general data in training. The resulting generic and domain-adapted systems based on Google Translate and those trained from scratch were evaluated on out-of-domain and in-domain data. The results of automatic evaluation mostly favour the from-scratch systems, and we further note that the performance of domain-adapted Google-based system drops for the out-of-domain data, whereas oversampling does not hurt the system's general performance. This is of minor importance when adaptation to a single domain is required, but might be of interest for companies that would want the MT system to also perform satisfactorily for other domains.

To get a better understanding of the system's performance, the in-domain dataset was also evaluated by a group of human evaluators of different backgrounds who all had some experience with medical translation. The participants were asked to evaluate the systems on accuracy and usability, and overall the generic Google Translate and the domain-adapted from-scratch system were judged to perform best on both criteria. However, due to their high variability, the results were inconclusive, which leads us to claim that all of the examined systems present viable options for domain-adapted MT.

Acknowledgements

We thank Cochrane for providing us with their EN-HR translation memory, and our evaluators and aligners for their time and effort.

References

Arčan, Mihael, Daniel Torregrosa, and Paul Buitelaar. 2017. Translating Terminological Expressions in Knowledge Base s with Neural Machine Translation. https://arxiv.org/pdf/1709.02184.pdf [last accessed September 30, 2019].

Axelrod, Amittai, Xiaodong He, and Jianfeng Gao. 2011. Domain Adaptation via Pseudo In-Domain Data Selection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 355-362.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the 3rd International Conference on Learning Representations*.

Castilho, Sheila, Natália Resende, Federico Gaspari, Andy Way, Tony O'Dowd, Marek Mazur, Manuel Herranz, Alex Helle, Gema Ramírez-Sánchez, Víctor Sánchez-Cartagena, Mārcis Pinnis, and Valters Šics. 2019. Large-scale Machine Translation Evaluation of the iADAATPA Project. In *Proceedings of MT Summit XVII, Volume 2: Translator, Project and User Tracks*, pages 179-185.

Chen, Mia Xu, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2018. The Best of Both Worlds: Combining Recent Advances in Neural Machine Translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*, pages 76–86.

Chu Chenhui, Raj Dabre, and Sadao Kurohashi. 2017. An Empirical Comparison of Simple Domain Adaptation Methods for Neural Machine Translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Volume 2: Short Papers*, pages 385–391.

Chu, Chenhui, and Rui Wang. 2018. A Survey of Domain Adaptation for Neural Machine Translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319.

Cochrane Collaboration. 2019. About us. https://www.cochrane.org/about-us [last accessed September 30, 2019].

Etchegoyhen, Thierry, Anna Fernández Torné, Andoni Azpeitia, Eva Martínez Garcia, and Anna Matamala. 2018. Evaluating Domain Adaptation for Machine Translation Across Scenarios. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, pages 6-15.

Freitag, Markus, and Yaser Al-Onaizan. 2016. Fast Domain Adaptation for Neural Machine Translation. https://arxiv.org/pdf/1612.06897.pdf [last accessed September 30, 2019].

Google Cloud. 2019a. AutoML Translation pricing. https://cloud.google.com/translate/#pricing [last accessed September 30, 2019].

Google Cloud. 2019b. Cloud AutoML. https://cloud.google.com/automl/ [last accessed September 30, 2019].

Google Cloud. 2019c. Google Cloud Platform Free Tier. https://cloud.google.com/free/docs/gcp-free-tier [last accessed September 30, 2019].

Hashimoto, Kazuma, Akiko Eriguchi, and Yoshimasa Tsuruoka. 2016. Domain Adaptation and Attention-Based Unknown Word Replacement in Chinese-to-Japanese Neural Machine Translation. In *Proceedings of the 3rd* Workshop *on Asian Translation*, pages 75–83.

Hieber, Felix, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2017. Sockeye: A Toolkit for Neural Machine Translation. https://arxiv.org/pdf/1712.05690.pdf [last accessed September 30, 2019].

Imankulova, Aizhan, Raj Dabre, Atsushi Fujita, and Kenji Imamura. 2019. Exploiting Out-of-Domain Parallel Data through Multilingual Transfer Learning for Low-Resource Neural Machine Translation. In *Proceedings of MT Summit XVII, Volume 1*, pages 128-139.

Jean, Sébastien, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On Using Very Large Target Vocabulary for Neural Machine Translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing,* pages 1–10.

Klein, Guillaume, Yoon Kim, Yuntian Deng, Vincent Nguyen, Jean Senellart, and Alexander M. Rush. 2018. OpenNMT: Neural Machine Translation Toolkit. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas,* pages 177–184.

Kobus, Catherine, Josep Maria Crego, and Jean Senellart. 2017. Domain Control for Neural Machine Translation. In *Proceedings of Recent Advances in Natural Language Processing*, pages 372–378.

Koehn, Philipp, and Rebecca Knowles. 2017. Six Challenges for Neural Machine Translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, Companion Volume, Proceedings of the Demo and Poster Sessions*, pages 177–180.

Luong, Minh-Thang, and Christopher D. Manning. 2015. Stanford Neural Machine Translation Systems for Spoken Language Domains. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 76-79.

Luong, Thang, Hieu Pham and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311-318.

Pighin, Daniele, Lluís Formiga, and Lluís Màrquez. 2012. A Graph-based Strategy to Streamline Translation Quality Assessments. In *Proceedings of the 10th Conference of the Association for Machine Translation in the Americas*.

Popović, Maja. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392-395.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016a. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 86-96.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016b. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725.

Servan, Christophe, Josep Maria Crego, and Jean Senellart. 2016. Domain Specialization: A Post-training Domain Adaptation for Neural Machine Translation. https://arxiv.org/pdf/1612.06141.pdf [last accessed October 8, 2019]

Silva, Catarina. 2019. Improving Domain Adaptation for Machine Translation with Translation Pieces. In *Proceedings of Machine Translation Summit XVII, Volume 2: Translator, Project and User Tracks*, pages 204–212.

Šoštarić, Margita. 2018. Advanced fuzzy matching in the translation of EU texts. *Hieronymus, Journal of Translation Studies and Terminology 5*, pages 26-71.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 223-231.

Stanojević, Miloš, Amir Kamran, Philipp Koehn, and Ondřej Bojar. 2015. Explorer Results of the WMT 15 Metrics Shared Task. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 256–273.

Straka, Milan, and Jana Straková. 2017. Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88-99.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Proceedings of the 31st Conference on Neural Information Processing Systems*, pages 6000-6010.

Wang, Rui, Andrew Finch, Masao Utiyama, and Eiichiro Sumita. 2017a. Sentence Embedding for Neural Machine Translation Domain Adaptation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 560–566.

Wang, Rui, Masao Utiyama, Lemao Liu, Kehai Chen, and Eiichiro Sumita. 2017b. Instance Weighting for Neural Machine Translation Domain Adaptation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1482–1488.