Training and Adapting Multilingual NMT for Less-resourced and Morphologically Rich Languages

Matīss Rikters, Mārcis Pinnis, Rihards Krišlauks
Tilde
Vienības gatve 75A, Riga, Latvia
{matiss.rikters, marcis.pinnis, rihards.krislauks}@tilde.lv

Abstract

In this paper, we present results of employing multilingual and multi-way neural machine translation approaches for morphologically rich languages, such as Estonian and Russian. We experiment with different NMT architectures that allow achieving state-of-the-art translation quality and compare the multi-way model performance to one-way model performance. We report improvements of up to +3.27 BLEU points over our baseline results, when using a multi-way model trained using the transformer network architecture. We also provide open-source scripts used for shuffling and combining multiple parallel datasets for training of the multilingual systems.

Keywords: neural machine translation, multilingual machine translation, morphologically rich languages

1. Introduction

One of the major advantages of neural machine translation (NMT) is that unlike statistical machine translation (SMT), which was the previous industry standard (and is still actively used in commercial applications), NMT is trained and used jointly as a single end-to-end system without the need to optimize multiple independent models and relations between the models. However, training NMT systems for individual language pairs has shown to take significantly more time (e.g., two to three weeks or up to a week with newer platforms, such as Marian (Junczys-Dowmunt et al., 2016) or Google's Tensor2Tensor toolkit [1]) than training of SMT systems (e.g., less than a day or up to several days for large systems). But even with this advantage, using the traditional approaches, one would still need to train a separate model for each translation direction. Since running a high amount of GPU-intensive NMT models in a production environment can quickly sum up to an enormous resource-usage cost, it has been natural (as shown by related work in Section 2) to look for solutions that allow compressing the models into an even more dense end-to-end solution that is able to handle multiple languages and language pairs simultaneously.

Another benefit of a single model for multiple translation directions could be the ability to learn not just from the training data of the language pair in question, but also from language pairs that include one of the languages. The advantages of learning from multiple translation directions at the same time can be (1) the ability for a model to learn how to translate language specific attributes that are common to multiple languages at the same time, and (2) to learn and generalize translations that may not occur in the parallel corpus of, e.g., A↔B, but do occur in parallel corpora of, e.g., A↔C and C↔B and therefore are deducible.

This work has been driven by the need to identify the best neural network architectures for the development of one-way and multi-way NMT systems for low-resource language pairs that can be applied for low-resource NMT system development (and/or system adaptation) within the project "Forest Industry Communication Technologies".

The structure of this paper is as follows: Section 2 summarizes related work in multilingual and multi-way NMT; Section 3 introduces the setup of our experimental environment and data used; Section 4 outlines the main results in translation quality as well as speed and resource usage, and in Section 5 we look at several examples how translations produced by one-way systems differ from multi-way system translations. Finally, we conclude the paper in Section 6 and introduce plans for future work.

2. Related Work

Multilingual NMT has recently been investigated by several research groups. For instance, Firat et al. (2017) modify the current state-of-the-art attentional NMT approach by supplementing it with the ability to learn from multiple language pairs and multiple translation directions at the same time. They are able achieve this by creating a shared attention mechanism across the involved resources. The authors report improvements in translation quality over most individual baselines, using a single multilingual model trained on five language pairs in both directions. The authors especially highlight that by combining data from language pairs with many resources with data from a low-resource language pair, the quality gains for the low-resource language pair are higher.

Johnson et al. (2016) introduce a simple method for training a single-model multilingual NMT system, which does not require any modifications to the architecture of the system. They achieve this by adding a target language identifying token in the beginning of each source sentence of the training data. While they only report comparable and not outperforming results for models trained on high-resource language pairs, the biggest improvements are achieved in low-resource and even zero-shot translation. An interesting aspect of this approach is that, when trained on many translation directions at once, the same input sentence can be translated into any supported target language by changing only the target language identifying token.

[1] T2T: Tensor2Tensor Transformers - `https://github.com/tensorflow/tensor2tensor`

Ha et al. (2016) use a similar approach to Johnson et al. (2016) by only modifying training data and using the same NMT system architecture. The main difference is that they add a language identifying token to each subword unit and apply this pre-processing to both - source and target sentences of the training data. Another difference is that they don't use particularly deep network architectures in their experiments. The authors describe two experiment scenarios where they train systems to translate from multiple source languages into one target language by (1) adding an additional parallel corpus and (2) adding a monolingual corpus as the additional source and target data. The achieved improvements reach up to 2.6 BLEU points for the first approach and up to 3.15 BLEU points for the second approach.

3. Experiment Setup

In our experiments, we mainly followed the path of Johnson et al. (2016) by not making any modifications to the network architecture and modifying only the data during training and inference. We did, however, experiment with different encoder and decoder cell types and add slight modifications to the data iterator module for it to automatically read the multilingual multi-way training data in equal batches for each translation direction and prepend the target language symbol at the beginning of each source sentence. Our recurrent neural network NMT systems were trained with Nematus (Sennrich et al., 2017) using four main configurations. For training of the NMT systems with convolutional neural networks and transformer networks, we used Sockeye (Hieber et al., 2017). All SMT systems were trained using using the Moses (Koehn et al., 2007) toolkit in the Tilde MT platform (Vasiļjevs et al., 2012). The details of the models are as follows:

- Recurrent neural network models
  - Maximum sentence length of 50;
  - Multiplicative long short-term memory (Krause et al., 2017) (MLSTM) shallow one-way (MLSTM-SU - the baseline model)
    - Encoder and decoder cell type – MLSTM (same as used by Pinnis et al. (2017));
    - A shared subword unit vocabulary (Sennrich et al., 2016) of 25,000 tokens;
  - Gated recurrent units (GRU)
    - Encoder and decoder cell type – GRU;
    - Shallow multilingual multi-way (GRU-SM)
      - 1-layer encoder and 1-layer decoder;
    - Deep - one-way (GRU-DU) and multilingual multi-way (GRU-DM)
      - 4-layer encoder and 4-layer decoder;
      - 2 GRU transition operations applied in the encoder layer; 4 GRU transition operations applied in the decoder layer; 2 GRU transition operations applied in decoder layers after the first layer;
      - Additional incremental training (Freitag and Al-Onaizan, 2016) after convergence of the GRU-DM model, using only parallel training and development data of a single translation direction;
- Fully convolutional neural network models - one-way (FConv-U) and multilingual multi-way (FConv-M)
  - Encoder and decoder cell type - convolutional neural network (CNN);
  - 15-layer encoder and 15-layer decoder;
  - Maximum sentence length of 128;
- Transformer neural network models - one-way (Transformer-U) and multilingual multi-way (Transformer-M)
  - Encoder and decoder cell type - transformer;
  - Maximum sentence length of 128;
  - 6-layer encoder with convolutional embeddings;
  - 6-layer transformer decoder;
  - Each block (self-attention or feed-forward network) is
    - Pre-processed with layer normalization;
    - Post-processed with dropout and a residual connection;
- SMT one-way models (SMT)
  - Word alignment performed using fast-align (Dyer et al., 2013);
  - 7-gram translation models and the 'wbe-msd-bidirectional-fe-allff' reordering models;
  - Language model trained with KenLM (Heafield, 2011);
  - Tuned using the improved MERT (Bertoldi et al., 2009).

Common parameters for all multilingual multi-way experiments:

- Multilingual training data was shuffled in equal batches per translation direction and with the target language identifier added before each sentence as described by Johnson et al. (2016).
- A shared subword unit vocabulary of 50 000 tokens was used.

For all one-way experiments we used a smaller shared subword unit vocabulary of 24 500 tokens.
All other parameters for the models were identical – we clip the gradient norm to 1.0 (Pascanu et al., 2013), use a dropout of 0.2 and trained the models with Adadelta (Zeiler, 2012). We used a word embedding of size of 500, and hidden layers of size 1024. All models were trained until they reached convergence on validation data.

Language pair	Before filtering (Total/Unique)	After filtering (Unique)
En ↔ Et	62.5M / 24.3M	18.9M
En ↔ Ru	60.7M / 39.2M	29.4M
Ru ↔ Et	6.5M / 4.4M	3.5M

Table 1: Training data sentence counts before and after filtering

3.1. Data

For training, we used English↔Russian, English↔Estonian, and Russian↔Estonian data. The one-way models were trained on English↔Estonian and Russian↔Estonian data while the multilingual multi-way models were trained on data from all three language pairs in both directions. The training corpora consist of multiple publicly available and proprietary datasets. Among the public datasets, the largest were the MultiUN (Chen and Eisele, 2012), DGT-TM (Steinberger et al., 2012), Open Subtitles (Tiedemann, 2009), Tilde MODEL (Rozis and Skadiņš, 2017), and Microsoft Translation Memories and UI Strings Glossaries (Microsoft, 2015). The corpora were cleaned and filtered in order to reduce noise in the parallel training data. During filtering, we removed non-parallel sentence pairs, sentences with sentence splitting errors, and duplicate entries. Data processing was performed in two steps – first, a low content overlap filter, which is based on the cross-lingual alignment tool *MPAligner* (Pinnis, 2013), was applied, followed by the standard data processing pipeline of the Tilde MT platform. For some corpora, the filtering resulted in an overall reduction of more than 50% of the original size. Corpora with content overlap below a certain threshold were manually examined and left out from the final dataset. The data filtering procedure is described in greater detail in the paper by Pinnis et al. (2017). An overview of the training data statistics before and after filtering for each language pair is given in Table 1.

For Estonian↔Russian, we selected 2000 random sentences from the training data to be used as validation data. The validation datasets for all other translation directions were obtained from the ACCURAT development datasets (Skadiņa et al., 2012). In the multilingual multi-way model training scenarios, we concatenated $\frac{1}{6}^{\text{th}}$ of each 2000 sentence validation dataset, resulting in batches of 333 sentences from each translation direction, which we used as development data. As for evaluation data – we used the ACCURAT balanced evaluation corpus (Skadiņš et al., 2010) consisting of 512 sentences in each translation direction, for which the Russian version was prepared by in-house translators.

4. Results

In this section, we describe the results of our experiments. We evaluate MT system translation quality using BLEU (Papineni et al., 2002). we also analyse translation speed and GPU memory usage during translation, as well as training duration. While training models for multiple translation directions, we were mainly focused on improving the translation quality when translating between Russian and Estonian, because this specific language pair had the poorest performance among the baseline systems.

4.1. Translation Quality

Table 2 shows how each of the models that we described in the previous section compares to the baseline in terms of development and evaluation data translation quality. When we compare the baseline one-way model (MLSTM-SU) to the other one-way models, the results show that the GRU-DU and FConv-U models reach lower translation quality on all development sets and all but one (for FConv-U) or two (for GRU-DU) evaluation sets. The GRU-DU model insignificantly out-performs the baseline model on the Estonian→Russian evaluation set (by 0.04 BLEU points) and the Estonian→English evaluation set (by 0.08 BLEU points). The FConv-U model shows slightly higher results (by 0.18 BLEU points) on the Estonian→English evaluation set. However, the results of the Transformer-U model are interesting. Although it got lower results on the Estonian↔Russian evaluation sets (by -1.15 and -2.01 BLEU points), it outperformed the baseline model on the Estonian↔Russian evaluation sets (by 2.29 and 3.3 BLEU points). A potential explanation of these results is that the Transformer-U model becomes more advantageous than the MLSTM-SU model when using larger data sets, however, for smaller datasets the MLSTM-SU model is still able to achieve state-of-the-art results.

Next, we look at whether the multi-way models allow increasing translation quality over one-way models. The results show that the GRU multi-way model outperforms the one-way models for all language pairs on all datasets. However, the convolutional and transformer models increase quality only for the low-resource language pairs. The quality improvement for the Estonian↔Russian language pairs ranges from 2.16 BLEU points (for the FConv-M model on the Estonian→Russian evaluation set) up to 5.28 BLEU points (for the Transformer-M model on the Russian→Estonian evaluation set). For the high-resource language pairs, on the other hand, both FConv-M and Transformer-M models show significantly lower translation quality than their respective one-way models. The quality decrease ranges from -2.11 BLEU points (for the Transformer-M model on the Estonian→English evaluation set) down to -5.17 BLEU points (for the FConv-M model on the Estonian→English evaluation set). This shows that the newer NMT architectures in multi-way scenarios are beneficial only to low-resource language pairs.

Finally, if we look at which models achieved the highest overall results on evaluation sets, it is evident that the transformer models performed the best. For the low-resource language pairs, the best results were achieved by the multi-way model. However, for the high-resource language pairs, the best results were achieved by the respective one-way models.

The reason why the results of the SMT system on the development set for Estonian↔Russian (underlined) are so much higher than for all other models may be due to the characteristic of SMT systems being good at memorizing similar sentences to what they have already seen during training.

	Development				Test			
	Ru → Et	Et → Ru	En → Et	Et → En	Ru → Et	Et → Ru	En → Et	Et → En
SMT	27.74	25.48	17.99	25.89	9.88	7.27	21.44	29.69
MLSTM-SU	17.51	18.46	23.79	34.45	11.11	12.32	26.14	36.78
GRU-SM	13.70	13.71	17.95	27.84	10.66	11.17	19.22	27.85
GRU-DU	17.03	17.42	23.53	33.63	10.33	12.36	25.25	36.86
GRU-DM	17.07	17.93	23.37	33.52	13.75	14.57	25.76	36.93
FConv-U	15.24	16.17	21.63	33.84	7.56	8.83	24.87	36.96
FConv-M	14.92	15.80	18.99	30.25	10.65	10.99	21.65	31.79
Transformer-U	17.44	18.90	**25.27**	**37.12**	9.10	11.17	**28.43**	**40.08**
Transformer-M	**18.03**	**19.18**	23.99	35.15	**14.38**	**15.48**	25.56	37.97

Table 2: Translation quality results for all model architectures on development and evaluation data. The best results are in bold.

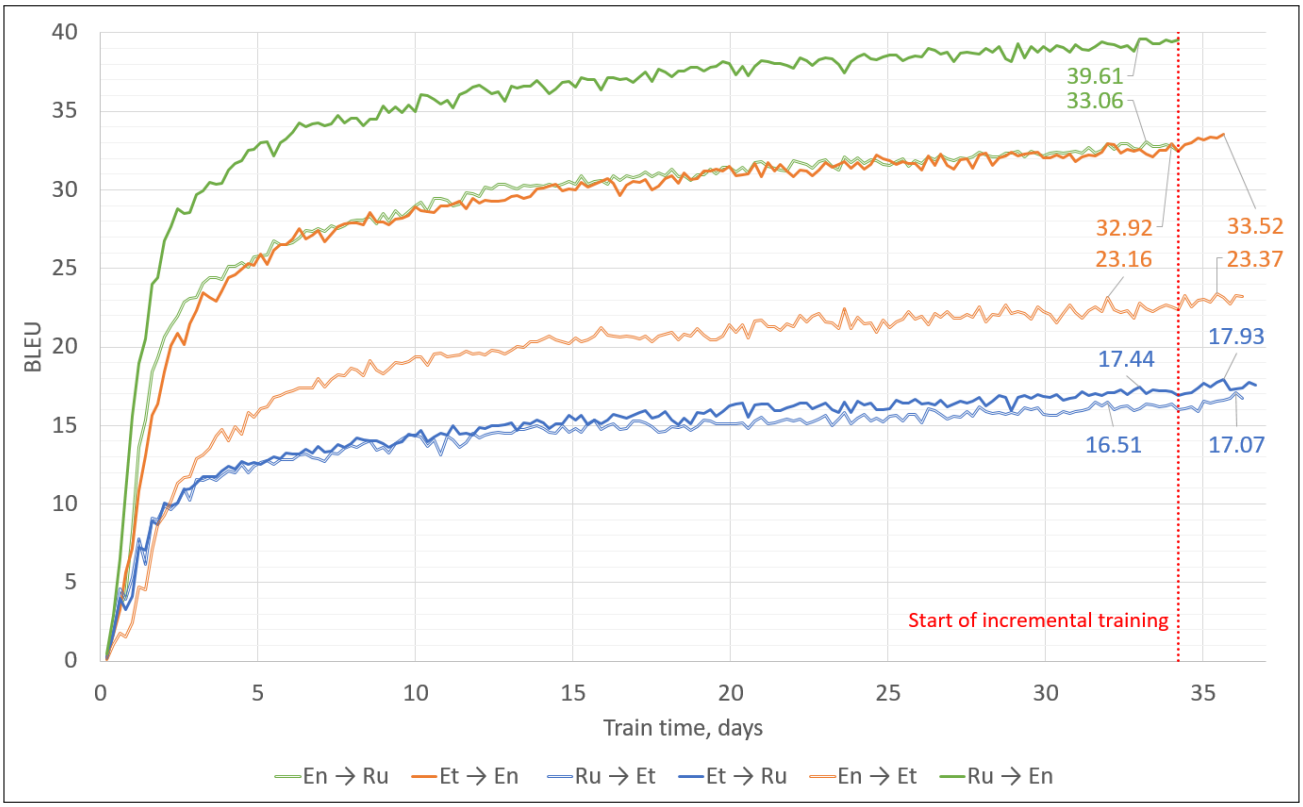


Figure 1: Training progress for the deep multilingual multi-way model (GRU-DM).

As stated in the previous section, this was the only language pair for which the development dataset was derived from the training dataset. For all other language pairs, we used a separate dataset.

When the GRU-DM model had converged, we performed additional incremental training for two language pairs in both ways (English↔Estonian and Russian↔Estonian). Figure 1 illustrates the training progress of this model and the four individual incrementally trained models. The idea of the incremental training was to adapt the system to a specific domain, which in this case would be translation into a single language. Incremental training improved the translation quality of the multi-way GRU-DM model for the individual language pairs by up to 0.60 BLEU points.

Figure 2 shows the training progress for multiple variations of Russian↔Estonian models. The deep one-way models (Estonian↔Russian GRU-DU) reached the early stopping criterion very quickly, but did not get as high as the other models over more time. The other RNN-based models converged after observing approximately 142 million sentences during training. The transformer models stand out the most by being the very first to stop training, as well as reaching the highest BLEU scores the quickest.

4.2. Resource Usage During Translation

Training models with deeper architectures increases resource usage in both – training time and required computational power. The higher resource usage is present during translation as well. Table 3 shows a comparison of time and GPU RAM consumption when translating the evaluation dataset using the NMT systems with several architectures from our experiments. In the table, we isolate models trained with Nematus from models trained with Sockeye, as they are based on different deep learning frameworks, re-

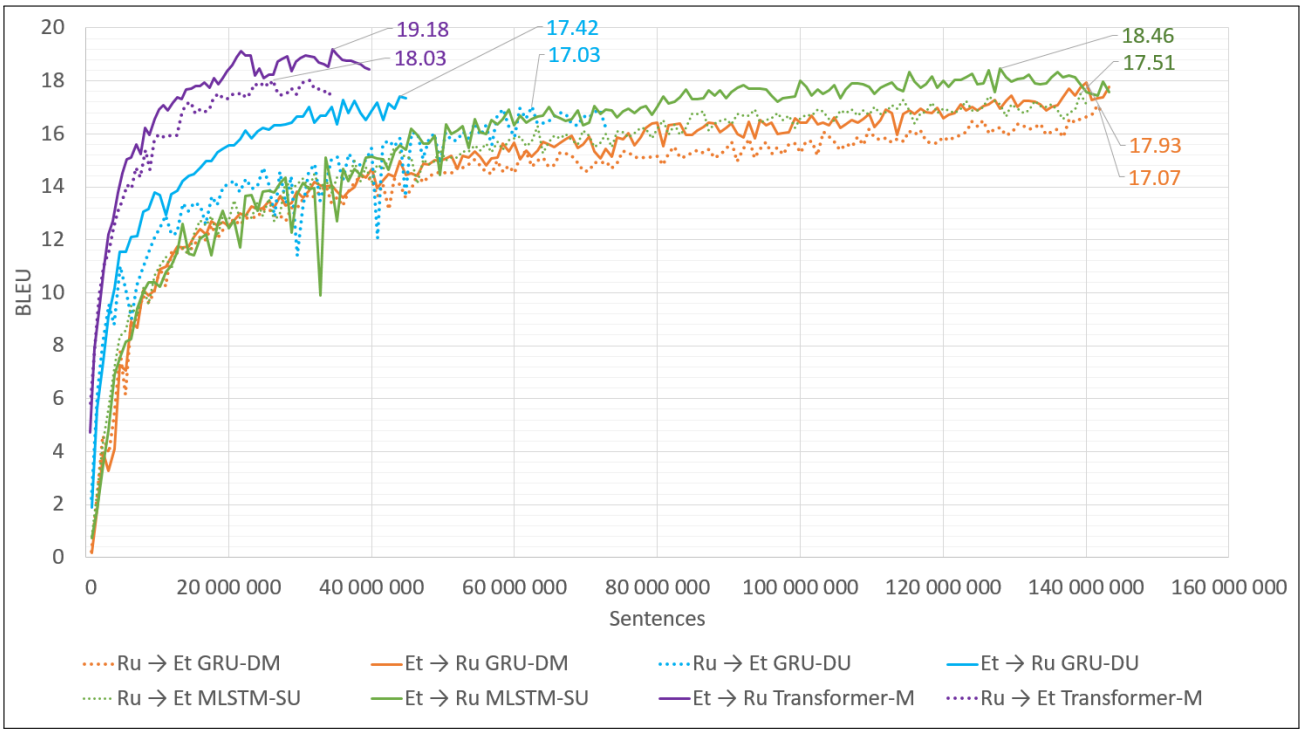


Figure 2: Training progress for Russian↔Estonian systems

	Seconds		Sentences	GPU RAM,	Train time,
	Translation	Per sentence	per second	MB	days
Theano-based Nematus					
MLSTM-SM	274.57	0.54	1.86	651	16.4
GRU-SM	211.51	0.41	2.42	**611**	8.5
GRU-DM	460.07	0.90	1.11	979	36.6
MXNet-based Sockeye					
FConv-M	177.19	**0.35**	**2.89**	971	4.5
Transformer-M	191.05	0.37	2.68	1391	**3.8**

Table 3: Resource usage for all NMT model architectures during translation. The most efficient values are in bold. The final column shows the training time until the system converges.

spectively, Theano (Theano Development Team, 2016) and MXNet (Chen et al., 2015).

The highest-scoring Transformer models are the quickest to train and also nearly the fastest during translation, but they consume more than twice the amount of GPU memory during translation. The GRU-DM model, which was the runner-up model for translating Estonian↔Russian uses 30% less GPU memory during translation, but takes 2.4 times longer to complete the job, and training also took 50% longer. All tests were performed on a machine with an NVIDIA Titan X (Pascal) GPU, Intel Core i7-6850K CPU @ 3.60GHz, 64GB of RAM, and 1TB SSD. We only used a single GPU for training and translating, even though the frameworks have support for multi-GPU training and translation.

It is worth mentioning that while training all shallow RNN models – multi-way or one-way – the training time for a single model to converge did not change noticeably. The same can be said about CNN and Transformer models. In the case of deep RNN models, training time increased by about 2-3 times, reaching 3-4 weeks on a single GPU.

5. Translation Examples

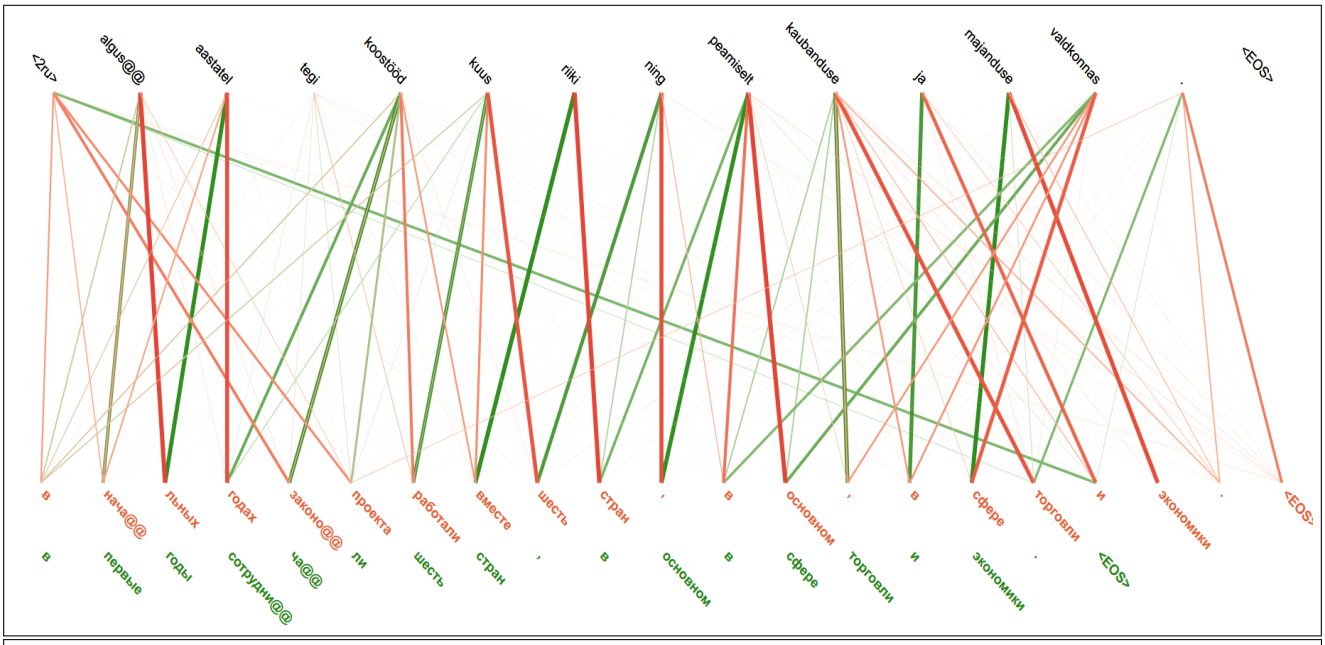
In this section, we show three examples where we compare sentences from one-way and multi-way architectures (e.g. the deep GRU models or transformer models).

In Figure 3, we compare one of the poorest-scoring translations generated with both the overall highest-scoring multi-way system (Transformer-M) and its one-way counterpart. The BLEU score of both translations is identical, but while the translation of Transformer-M is almost perfect (with fluency issues in the last two words), the translation of Transformer-U features a more significant lexical choice mistake. I.e., the words "*kasutab*" (uses) and "*regulaarselt*", which are correctly translated by the multi-way model as "использует" (uses) and "регулярно" (regularly), are mistranslated by the one-way model as "практикуют" (practice) and "работу" (work).

Figure 4 shows a comparison of a sentence that had one of the highest BLEU scores out of all GRU-DU translations compared with the same sentence translated using GRU-DM. There is a redundant word ("законопроекта" - bill project or draft law) in the translation of the one-way

Source:	Üle poole rahvastikust kasutab Internetti regulaarselt.
Transformer-U:	более половины населения практикуют работу с Интернетом.
(transl. into English):	*More than half of the population practice working with the Internet.*
Transformer-M:	более половины населения регулярно использует Интернет.
(transl. into English):	*More than half of the population regularly uses the Internet.*
Reference:	более половины жителей регулярно пользуются интернетом.
English Reference:	More than half the population are regular internet users.

Figure 3: Translation examples comparing the highest-scoring system (multi-way transformer) with its one-way counterpart. BLEU score of both - **15.62**.



Source:	Algusaastatel tegi koostööd kuus riiki ning peamiselt kaubanduse ja majanduse valdkonnas.
GRU-DU:	в начальных годах законопроекта работали вместе шесть стран , в основном , в сфере торговли и экономики.
(transl. into English):	*In the initial years of the bill project, six countries worked together, mainly in the sphere of trade and economy.*
GRU-DM:	в первые годы сотрудничали шесть стран , в основном в сфере торговли и экономики.
(transl. into English):	*In the first years, six countries cooperated, mainly in the sphere of trade and economy.*
Reference:	в первый год сотрудничество вели шесть стран , в основном в сфере торговли и экономики.
English Reference:	In the early years , the cooperation was between six countries and mainly about trade and the economy.

Figure 4: Translation examples comparing the second highest-scoring system (deep multi-way GRU) with its one-way counterpart. BLEU scores - **47.63** (GRU-DU - orange alignments) and **67.04** (GRU-DM - green alignments).

Source:	Charles tõusis ja vaatas aknast välja.
Transformer-U:	Шарль встал и посмотрел в окно.
(transl. into English):	*Charles stood up and looked out the window.*
Transformer-M:	Шарль встал и оглянулся в окно.
(transl. into English):	*Charles stood up and looked out the window.*
Reference:	Чарльз поднялся и посмотрел в окно.
English Reference:	Charles rose and looked out of the window.

Figure 5: Translation examples comparing the highest-scoring system (multi-way transformer) with its one-way counterpart. BLEU scores - **61.48** (Transformer-U) and **26.27** (Transformer-M).

model, which is not present in the source. It is also evident in the attention alignments (visualised using the toolkit by Rikters et al. (2017)) that the sub-word units of this word are strongly aligned only to the target language tag at the beginning of the source sentence. This may mean that these are not translations of any specific sub-word units of the source sentence. The translation of the multi-way model does not exhibit such a problem in this example.

In Figure 5, we show the third example. Here the translation from the one-way transformer model scores higher according to BLEU than the multi-way model. The only difference between these two translations is how the Estonian word "*vaatas*" (looked) is translated. The Transformer-U model produced the translation "посмотрел" (looked), which matches the reference translation, but the Tranformer-M model produced the translation "оглянулся" (looked back), which is the wrong lexical choice in the given context.

6. Conclusion

In this paper, we described a wide range of experiments on training and evaluating multilingual and multi-way neural machine translation systems. Our results show that for low-resource language pairs, such as Estonian↔Russian, we can achieve a significant improvement in translation quality by adding data from other languages over using only one-way parallel data. Multi-way NMT systems in both directions improved translation quality (by 3.09 - 5.28 BLEU points for Russian→Estonian and 2.16 - 4.31 BLEU points for Estonian→Russian) for all three model architectures (deep GRU, convolutional, and transformer), for which we performed multi-way experiments. Our experiments also show that the largest improvements in BLEU scores, as well as the highest overall BLEU scores in the low-resource multi-way scenario were achieved by training systems with the Transformer model.

While the multilingual approach helped gaining improvements for the low-resource language pair, it did degrade the performance for the high-resource language pairs by several BLEU points. In almost all of our experiments the multilingual models showed a drop in translation quality by 2.87 - 3.22 BLEU points for English→Estonian and 2.11 - 5.17 BLEU points for Estonian→English. However, the results showed that the most stable architecture for multi-way model training was the deep GRU model architecture. It showed improvements for both low-resource and high-resource language pairs on both development and evaluation data sets.

The results also showed that when training one-way systems for the low-resource language pairs, the newer convolutional and self-attention (i.e., transformer) models underperformed. The best results in these experiments were achieved by the MLSTM-based models (outperforming the convolutional models by up to 3.55 BLEU points and the transformer model by 2.01 BLEU points).

While manually analysing the evaluation sets, we noticed that there were several sentences translated perfectly by Transformer-M, but much worse by GRU-DM and vice versa. This suggests that further investigation may be required to find out whether a combination of the systems can lead to translations of even higher quality. There are many successful methods for MT system combination that could be utilized, for example, using confusion networks (Peter et al., 2017) to align hypotheses and pick the best parts of each as the final translation. A more neural network specific option for MT system combination by combining outputs according to the attention alignments produced by the neural networks (Rikters and Fishel, 2017) could also be used for this purpose.

Finally, we provide an update to Nematus[2] that allows training of multi-way models by providing multiple parallel corpora as input data. We also release a set of scripts[3] that can be used to prepare a multi-way corpus from multiple parallel corpora for training of multi-way NMT systems with other frameworks.

7. Acknowledgements

In accordance with the contract No. 1.2.1.1/16/A/009 between the "Forest Sector Competence Centre" Ltd. and the Central Finance and Contracting Agency, concluded on 13th of October, 2016, the study is conducted by Tilde Ltd. with support from the European Regional Development Fund (ERDF) within the framework of the project "Forest Sector Competence Centre".

8. Bibliographical References

Bertoldi, N., Haddow, B., and Fouet, J.-B. (2009). Improved Minimum Error Rate Training in Moses. *The Prague Bulletin of Mathematical Linguistics*, 91(1):7—-16.

Chen, Y. and Eisele, A. (2012). MultiUN v2: UN Documents with Multilingual Alignments. In *LREC*, pages 2500–2504.

Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., and Zhang, Z. (2015). MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems. *Neural Information Processing Systems, Workshop on Machine Learning Systems*, pages 1–6.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. Association for Computational Linguistics.

Firat, O., Cho, K., Sankaran, B., Yarman Vural, F. T., and Bengio, Y. (2017). Multi-way, Multilingual Neural Machine Translation. *Computer Speech and Language*, 45:236–252.

Freitag, M. and Al-Onaizan, Y. (2016). Fast Domain Adaptation for Neural Machine Translation. *arXiv [cs.CL]*.

Ha, T.-L., Niehues, J., and Waibel, A. (2016). Toward Multilingual Neural Machine Translation with Universal Encoder and Decoder. In *Proceedings of the 13th International Workshop on Spoken Language Translation (IWSLT 2016)*, page 16.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on*

[2]Multilingual NMT iterator - git.io/vAgfv

[3]Multilingual NMT Corpora Tools - git.io/vAOoJ

Statistical Machine Translation, pages 187–197. Association for Computational Linguistics.

Hieber, F., Domhan, T., Denkowski, M., Vilar, D., Sokolov, A., Clifton, A., and Post, M. (2017). Sockeye: A Toolkit for Neural Machine Translation. *ArXiv e-prints*, dec.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., et al. (2016). Google's Multilingual Neural Machine Translation System: Enabling Zero-shot Translation. *arXiv preprint arXiv:1611.04558*.

Junczys-Dowmunt, M., Dwojak, T., and Hoang, H. (2016). Is Neural Machine Translation Ready for Deployment? A Case Study on 30 Translation Directions. In *Proceedings of the 9th International Workshop on Spoken Language Translation (IWSLT)*, Seattle, WA.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Krause, B., Lu, L., Murray, I., and Renals, S. (2017). Multiplicative LSTM for sequence modelling. In *5th International Conference on Learning Representations*, page 9, Toulon, France, feb.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. *... of the 40Th Annual Meeting on ...*, pages 311–318.

Pascanu, R., Mikolov, T., and Bengio, Y. (2013). On the Difficulty of Training Recurrent Neural Networks. In *International Conference on Machine Learning*, pages 1310–1318.

Peter, J.-T., Ney, H., Bojar, O., Pham, N.-Q., Niehues, J., Waibel, A., Burlot, F., Yvon, F., Pinnis, M., Sics, V., et al. (2017). The QT21 Combined Machine Translation System for English to Latvian. In *Proceedings of the Second Conference on Machine Translation*, pages 348–357.

Pinnis, M., Krišlauks, R., Miks, T., Deksne, D., and Šics, V. (2017). Tilde's Machine Translation Systems for WMT 2017. In *Proceedings of the Second Conference on Machine Translation (WMT 2017), Volume 2: Shared Task Papers*, pages 374–381, Copenhagen, Denmark. Association for Computational Linguistics.

Pinnis, M. (2013). Context Independent Term Mapper for European Languages. In *Proceedings of the 9th International Conference on Recent Advances in Natural Language Processing (RANLP 2013)*, Hissar, Bulgaria.

Rikters, M. and Fishel, M. (2017). Confidence Through Attention. In *Proceedings of the 16th Machine Translation Summit (MT Summit 2017)*, Nagoya, Japan.

Rikters, M., Fishel, M., and Bojar, O. (2017). Visualizing Neural Machine Translation Attention and Confidence. volume 109, pages 1–12, Lisbon, Portugal.

Rozis, R. and Skadiņš, R. (2017). Tilde MODEL-Multilingual Open Data for EU Languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 263–265.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli Barone, A. V., Mokry, J., and Nadejde, M. (2017). Nematus: a Toolkit for Neural Machine Translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain, April. Association for Computational Linguistics.

Skadiņa, I., Aker, A., Mastropavlos, N., Su, F., Tufiș, D., Verlic, M., Vasiļjevs, A., Babych, B., Clough, P., Gaizauskas, R., Glaros, N., Paramita, M. L., and Pinnis, M. (2012). Collecting and Using Comparable Corpora for Statistical Machine Translation. In Nicoletta Conference Chair Calzolari, et al., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 438–445, Istanbul, Turkey. European Language Resources Association (ELRA).

Skadiņš, R., Goba, K., and Šics, V. (2010). Improving SMT for Baltic Languages with Factored Models. In *Frontiers in Artificial Intelligence and Applications*, volume 219, pages 125–132.

Steinberger, R., Eisele, A., Klocek, S., Pilos, S., and Schlter, P. (2012). DGT-TM: a Freely Available Translation Memory in 22 Languages. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 454–459.

Theano Development Team. (2016). Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May.

Tiedemann, J. (2009). News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In *Recent advances in natural language processing*, volume 5, pages 237–248.

Vasiļjevs, A., Skadiņš, R., and Tiedemann, J. (2012). LetsMT!: A Cloud-Based Platform for Do-It-Yourself Machine Translation. In Min Zhang, editor, *Proceedings of the ACL 2012 System Demonstrations*, number July, pages 43–48, Jeju Island, Korea. Association for Computational Linguistics.

Zeiler, M. D. (2012). ADADELTA: an Adaptive Learning Rate Method. *arXiv preprint arXiv:1212.5701*.

9. Language Resource References

Microsoft. (2015). *Translation and UI Strings Glossaries*.