Baltic J. Modern Computing, Vol. 7 (2019), No. 3, 301-341
https://doi.org/10.22364/bjmc.2019.7.3.01

Hybrid Machine Translation by Combining Output from Multiple Machine Translation Systems

Matīss RIKTERS

Faculty of Computing, University of Latvia, Ranis Blvd. 19, Riga, LV-1586, Latvia

matiss.rikters@lu.lv

Abstract: This paper aims to combine output from various machine translation (MT) systems so that the overall translation quality of the source text would increase. Applicability of the developed methods for small, morphologically rich and under-resourced languages is evaluated, especially Latvian and Estonian. Existing methods have been analysed, and several combinations of methods have been proposed. The proposed methods have been implemented and evaluated using automatic and human evaluation. During this research novel methods have been created that structure source language sentences into linguistically motivated fragments and combine them using a character level neural language model; combine neural machine translation output by employing source-translation attention alignments; use a multi-pass approach to produce additional incrementally improving training data. The key results of this research are new state-of-the-art machine translation systems for English ↔ Estonian; approaches for utilising neural MT generated attention alignments for MT combination and comprehension of resulting translations; MT combination systems for combining output from English → Latvian statistical MT. A practical application of the methods is implemented and described.

Keywords: Machine Translation, Hybrid Machine Translation, Machine Translation System Combination, Multi-System Machine Translation

1. Introduction

Today most commercial MT systems are built using a variety of statistical approaches and the most recent - neural network-based neural MT (NMT) approaches. Currently MT has not yet reached a level of quality where it can entirely replace a human translator, and it probably will not reach this level in any near future. However, MT has become highly useful in scenarios such as providing an initial translation for post-editing or extracting information from texts in foreign languages. In the digital age of our multicultural world, the demand for faster and cheaper translation has breed many commercial products (e.g. IBM WebSphere Translation Server, Systran, SDL BeGlobal) and multiple translation services are freely available on the web or as mobile applications (e.g. Google Translate [1], Bing Translator [2], Yandex.Translate [3], Baidu

[1] https://translate.google.com
[2] https://www.bing.com/translator
[3] https://translate.yandex.com

Translate[4], Tilde Translator[5]), demonstrating high translation quality for a wide variety of languages.

A lot of current research focuses on MT for the widely-used languages, like English, Chinese, Spanish, Portuguese, French, Arabic, Japanese and Russian, as well as languages that appear in competition shared tasks, like Czech, Finnish and Turkish. Much less work is being done in the area of hybrid methods, for instance, combining multiple different paradigms to utilise their strengths and cover weaker points. Smaller languages like the Baltic three - Estonian, Latvian and Lithuanian are far less resourced in available MT services, or even language technologies in general. The existing services and technologies for these languages lack sophistication due to little available linguistic resources and technological approaches that enable development of cost-effective MT. This has caused a technological gap to emerge between the two groups of languages.

Some systems like Google Translate, Bing Translator, Yandex Translator and Baidu Translate are freely available as online services and broaden the set of inter-translatable language pairs, even incorporating the languages of the Baltic countries as well as many other less resourced languages. Typically, these online translation services are employed to translate short texts by occasional users. Another common use-case is the translation of websites and, most recently, social media posts.

1.1. Motivation of the research

Rule-based, statistical and neural MT methods all have both stronger points as well as some noticeable weaknesses. Rule-based MT (RBMT) systems can achieve a high-quality translation if they have a full set of the knowledge necessary. RBMT typically handles specific language phenomena like word agreements, inflections, long distance reordering, and long-distance dependency, etc. better and output of RBMT systems is predictable and consistent, making it easy to locate and correct translation errors. Unfortunately, real-world human languages are complex with many ambiguities and exceptions, as well as always changing as time moves forward. Advancing RBMT is too complicated and labour-intensive due to linguistic expertise and domain knowledge needed to create RBMT systems where the knowledge for one language pair in one domain typically is not reusable in another language pair or domain.

In contrast, SMT systems do not need manually written knowledge sets like dictionaries and rules - they usually consist of subcomponents that are trained and optimized for usage separately, but with the same sets of data. Knowledge is automatically learned by training statistical models on large datasets, which makes improving and adapting systems to new language pairs more flexible. SMT is more challenging for highly inflectional languages that have too many word forms, cases, etc. for all possible word form and sentence construction variants to appear in training data. Therefore, SMT still struggles with word agreements, inflections, long distance reordering, and long-distance dependencies. A large, high-quality parallel corpus is essential for corpus-based MT, but it is often unavailable for small and less popular languages.

[4] http://translate.baidu.com
[5] https://translate.tilde.com

Similar to SMT, NMT is also trained on a large amount of parallel data. It is computationally expensive for both training the models and using them to translate texts. Another big difference is that neural systems are usually trained end-to-end without any subcomponents. Some drawbacks of NMT include struggles in rare word translation and sometimes even a complete failure to translate all given source sentence words. In addition, since some NMT systems do translation in the character level and not the word level, they have a tendency to make up new words that may almost look real but in fact, do not exist. However, the advantages definitely are in generalization and handling inflections.

Given that all of the MT methods have their advantages and drawbacks, it is reasonable to try to combine results from different MT systems to fix the mistranslations produced by one system with the help of the other systems. In addition, given that the Latvian language is spoken only by 1.95 million people, has a complicated grammar, rich morphology and limited amount of qualitative data, purely data-driven methods may not be sufficient. Combining results from several approaches has the potential to produce a better final result.

1.2. Aim of the research

The focus of this research is the problem of combining output from multiple different machine translation systems to acquire one superior final translation. This is an area that, when perfected, can achieve ever better results with every other single MT method (used here as a component) that improves upon itself. This paper describes problematic areas related to machine translation, limitations of current MT methods and provides suggestions on how to combine translations to achieve better overall quality of MT.

The primary goal is to assemble a set of methods that would be able to improve the quality of MT output for the languages of the Baltic countries that are small, have a rich morphology and little resources available. These characteristics currently make them rather difficult to translate with the tools that are currently available.

The research primarily focuses on solving MT problems that are related to translating from and into Latvian. Nevertheless, the aim is to find such methods that may be applied to other languages as well.

For this research, the author has suggested the following **hypothesis:**

Combining output from multiple different MT systems makes it possible to produce higher quality translations for the languages of the Baltic countries than the output that is produced by each component system individually.

The goal of this research is to create a method for combining output from multiple MT systems that provides a higher overall translation quality. This goal encompasses all of the following major aspects:

- An analysis of RBMT, SMT and NMT methods as well as existing HMT and multi-system MT (MSMT) methods;
- Experiments with different methods for combining translations;
- MT quality evaluation;
- Applicability of methods for Estonian, Latvian, Lithuanian, and other less resourced languages;
- Practical applications of combining MT.

1.3. Outline of the Paper

The rest of this paper is structured as follows:

- Chapter 2 summarizes existing machine translation methods and outlines advantages and disadvantages for each approach, especially detailing related work in the area of hybrid MT and existing combinations MT approaches.
- Chapter 3 introduces methods for combining translations from multiple statistical MT engines. It is based on research conducted before the high rise in popularity of neural approaches. For each method, an overview and relevance to the aims of this research is given, followed by a description of evaluation methods used, as well as a detailed description of the experiments made.
- Chapter 0 gives an insight into combining translations from neural MT engines. The research described in this chapter was conducted in the transition period between statistical and neural approaches, which called for different methods to be explored. The structure is similar to the previous chapter.
- Chapter 5 introduces several practical implementations that incorporate the previously mentioned translation combination methods for both – statistical and neural approaches.
- Chapter 6 sums up conclusions of this research.

2. Background and related work

Since the very first appearances of MT in the mid-20th century, there have been several main paradigms that have shifted from one to the next over the years. The focus of MT research started mainly with a dominance of rule-based approaches that were later accompanied by corpus-based MT, and after that several hybrid approaches to MT have appeared as well. In the most recent years, neural network-based MT is rapidly starting to outperform other methods in most use-cases.

This chapter introduces four of the main MT paradigms in the order of increasing interest by researchers and enterprise users over the course of history. Section 2.1 gives an insight on how MT is evaluated, section 2.2 covers rule-based, section 2.3 – corpus-based, section 2.4 – hybrid, and section 2.5 – neural approaches to MT.

2.1. MT Evaluation

To understand if an automatic translation is good or not, it must be compared to what a human translator would be able to produce, given the same source. Manual human evaluation is the best for such a task, but it is expensive and impractical for performing on large amounts of texts on a regular basis. This creates a demand for automatic evaluation metrics that have a high correlation with human judgments. Some of the first successful and most popular metrics are BLEU (Papineni et al., 2002), TER (Snover et al., 2006) and METEOR (Banerjee and Lavie, 2005).

The bilingual evaluation understudy (BLEU) is currently the most used and most cited MT evaluation metric. The main idea of BLEU **Error! Reference source not found.** is to reward MT outputs that have many overlapping n-grams (where n ranges from 1 to 4) with professional human translations (n-gram precision - **Error! Reference**

source not found., where $Count_{clip}$(n–gram) is the count of n-gram matches between a candidate translation and a reference truncated to not exceed the largest count of that n-gram that is observed in the reference and Count(n-gram') is the total number of n-grams in the test corpus), while penalizing translations that are shorter than the human reference (brevity penalty - (1), where *c* is the length of the candidate translation and *r* is the length of the reference). BLEU scores **Error! Reference source not found.** are usually computed using 4-gram precision where N=4 and weights $w_n=\frac{1}{N}$. BLEU scores are represented on a scale of 0.00 to 1.00, where 1.00 is the best and 0.00 – the worst, and the final results are typically multiplied by 100. The current state-of-the-art MT systems tend to reach between 20 and 40 BLEU points, depending on the language pair, translation direction and domain in question. Unless stated otherwise, all BLEU scores reported in this paper will be calculated using the *multi-bleu.perl* script from the Moses toolkit (Koehn et al., 2007).

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n-gram \in C} Count_{clip}(n-gram)}{\sum_{C' \in \{Candidates\}} \sum_{n-gram' \in C'} Count(n-gram')} \quad (1)$$

$$BP = \begin{cases} 1 & if\ c > r \\ \exp(1 - r/c) & if\ c \leq r \end{cases} \quad (1)$$

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{4} w_n \log p_n\right) \quad (3)$$

2.2. Rule-based MT

Rule-based MT (RBMT) is often denoted as the classical approach to MT. It mainly relies on the semantic, syntactic and morphological rules of the source and target languages as well as large monolingual dictionaries for each language and a bilingual dictionary for the actual translation between words. Most of this linguistic information needs to be composed by expert linguists, making it more expensive to build and expand if necessary. Some advantages of RBMT are complete control and ease of debugging, no need of large parallel corpora of texts, domain independence in many cases, and a certain level of reusability, for instance when using the same source language to translate into new target languages.

2.3. Corpus-based MT

Corpus-based MT uses large bilingual parallel text corpora as its primary resource. These corpora are used to train models for translation. Usually, the same setup can be used to train MT systems for multiple language pairs just by changing the training dataset thereby attempting to eliminate one of the general shortcomings of RBMT. One of the drawbacks is that while for the big and widely used languages the necessary corpora can be gathered in sufficient quantities, for smaller, less-used languages these corpora are often limited in size or non-existent at all.

One of the main corpus-based methods is Statistical MT (SMT). SMT produces translations according to the probability distribution of words in the target language (e.g. English) are translations of sentences in the source language (e.g. French). One approach

to modelling this probability distribution is to apply the Bayes Theorem, where the translation model (TM) calculates the probability that the target sentence is the translation of the source sentence, and the language model (LM) calculates the probability of seeing that sentence appear in the target language. Using these two models, a decoder performs the actual translation process.

2.4. Hybrid MT

Hybrid MT (HMT) represents a subset of MT where different MT approaches are used in the same system to complement each other's strengths in order to boost the accuracy level of the translation. Some of the well-known types of HMT include modifying SMT systems with RBMT generated output and generating rules for RBMT systems with the help of SMT. These systems would be categorized under the statistical rule generation subset of HMT. More recently NMT is used in combination with SMT (Marie et al., 2018). The other big subsets are multi-pass, where a sentence is fully translated with one MT system and the output is passed on as input for another MT system, and multi-system MT, where multiple translations of one sentence are generated in parallel.

2.5. Neural MT

NMT is the newest architecture for getting machines to learn to translate. NMT has shown promising results by achieving state-of-the-art performance for various language pairs (Sennrich et al., 2016). One of the main differences when compared to SMT methods, which consist of many small sub-components that are tuned separately, is that in NMT only one fully end-to-end model is trained and jointly tuned to maximize translation performance. Some drawbacks include a rather poor performance for long sentences, production of multiple repeated translations of a phrase and most notably – dealing with unknown words. These troubles have been addressed by shifting from word level translation to sub-word level or even character level translation, which introduced a new problem – the occasional production of new, non-existing words in the output translation. The first pure neural MT was introduced with encoder-decoder models (Sutskever et al., 2014; Cho et al., 2014) and later enhanced by adding attention (Bahdanau et al., 2015). More modern approaches use different neural network structures, such as convolutional neural networks (Gehring et al., 2017) or self-attentional models (Vaswani et al., 2017). Currently, most state-of-the-art systems for popular and well-resourced to medium-resourced language pairs are either some form of NMT or have NMT as a key component in a hybrid setup.

3. Combining statistical machine translation output

3.1. Combining full sentence translations

This section presents the first attempt in using an MSMT approach for the less-resourced English-Latvian language pair. The system consists of three major constituents – tokenization of the source text, the acquisition of a translation via online APIs and the selection of the best translation from the candidate hypotheses. A visualized workflow of

the system is presented in Figure 1. The system uses three translation APIs (Google Translate[6], Bing Translator[7] and LetsMT[8]). The section is based on the paper of Rikters (2015).

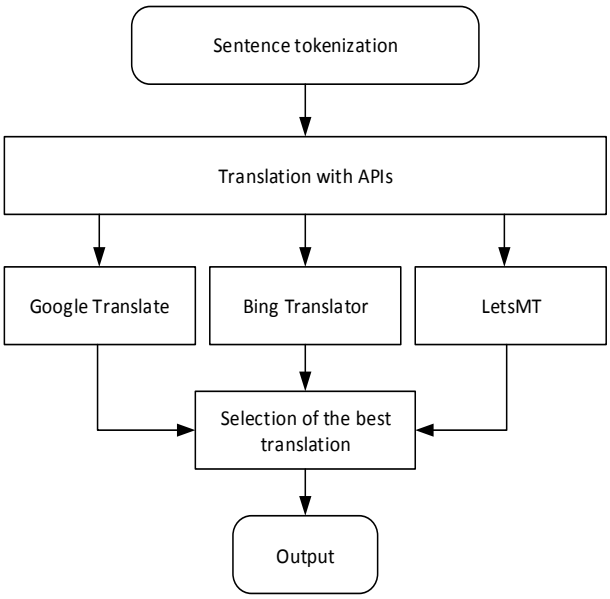


Figure 1. General workflow of the translation process

The best translation is selected by calculating the perplexity of each hypothesis translation using KenLM (Heafield, 2011). First, a language model (LM) must be created using a preferably large set of training sentences. Then, for each machine-translated sentence, a perplexity score represents the probability of the specific sequence of words appearing in the training corpus used to create the LM. Perplexity on a test set is calculated using the language model as the inverse probability (P) of that test set, which is normalized by the number of words (N) (Jurafsky and Martin, 2014). For a test set $W = w_1, w_2, ..., w_N$:

$$perplexity(W) = \mathrm{P}(w_1, w_2, \ldots, w_N)^{-\frac{1}{N}} = \sqrt[N]{\frac{1}{\mathrm{P}(w_1, w_2, \ldots, w_N)}} \quad (2)$$

Perplexity can also be defined as the exponential function of the cross-entropy:

[6] Google Translate API - https://cloud.google.com/translate/
[7] Bing Translator Control - http://www.bing.com/dev/en-us/translator
[8] LetsMT! Open Translation API - https://www.letsmt.eu/Integration.aspx

$$H(W) = -\frac{1}{N}\log \mathrm{P}(w_1, w_2, ??, w_N) \quad (3)$$

$$perplexity(W) = 2^{H(W)} \quad (4)$$

One set of experiments was conducted on the English – Latvian part of the JRC-Acquis corpus version 3.0 (JRC) (Steinberger et al., 2006) from which both the language model and the evaluation data were retrieved. The language model was created using KenLM with order 5. The results are summarized in

Table **2**. The combination of Google Translate and Bing Translator shows improvements in BLEU score and WER compared to each of the baseline systems. Since the systems themselves are more of a general domain and the first evaluation was conducted on a legal domain corpus, a second experiment (shown in Table 1) was conducted on a smaller general domain dataset (Skadiņa et al., 2010). The results showed that there is potential for this method of MT combination.

Table 1. Second experiment results on 512 general domain sentences.

System	BLEU
Google Translate	24.73
Bing Translator	22.07
LetsMT	32.01
Google + Bing	23.75
Google + LetsMT	28.94
LetsMT + Bing	27.44
Google + Bing + LetsMT	26.74

Table 2. First experiment results on 1581 random legal domain sentences from JRC.

System	BLEU	TER	WER
Google Translate	16.92	47.68	58.55
Bing Translator	17.16	49.66	58.40
LetsMT	28.27	36.19	42.89
Google + Bing	**17.28**	48.30	**58.15**
Google + LetsMT	22.89	41.38	50.31
LetsMT + Bing	22.83	42.92	50.62
Google + Bing + LetsMT	21.08	44.12	52.99

Table 3. Native speaker evaluation results.

System	AVG User	Hybrid	BLEU
Bing	31.88%	28.93%	16.92
Google	30.63%	34.31%	17.16
LetsMT	37.50%	33.98%	28.27

A random 2% (32 sentences) of the translations from the first experiment were given to five native Latvian speakers with an instruction to choose the best translation (just like the hybrid system should). The results in Table 3 show a tendency towards the LetsMT translation among the user ratings and BLEU score that is not visible from the selection of the hybrid method.

3.2. Combining sentence fragment translations - simple fragmenting

This section presents a method for improving the MSMT approach by incorporating syntactic information. One of the typical errors produced by SMT engines is wrong inflection (Skadiņa et al., 2012), which is usually caused by ignoring syntax rules. This approach attempts to improve the situation by translating smaller, linguistically motivated chunks of full sentences. The section is based on the paper of Rikters and Skadiņa (2016a).

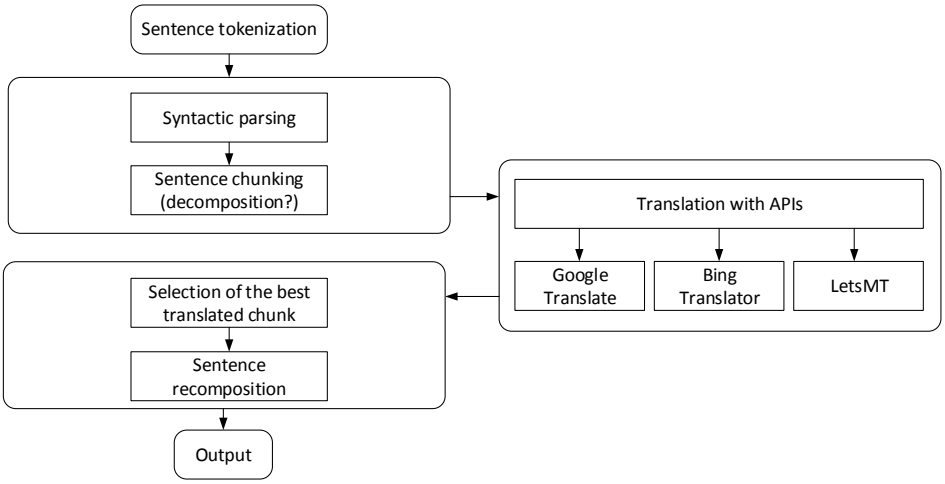


Figure 2. General workflow of the translation process.

The system consists of similar components to the one in the previous section. The main difference is the inclusion of syntactic components. A visualized workflow of the

system is presented in Figure 2. Prerequisites for compatible languages are support by the MT APIs for translation and the Berkeley Parser (Petrov et al., 2006) for syntactic analysis.

In order to divide sentences into chunks the parse tree of each sentence is processed by a chunk extractor to obtain the top-level sub-trees (noun phrases, verb phrases, prepositional phrases, etc.). This step relies only on source language parser and does not consider properties of the target language, i.e., it is independent of the target language. The selection of the best-translated chunk is performed as described in Section 3.1 for full sentences. Finally, the translation of the full sentence is obtained by concatenation of chunks.

Experiments were conducted using the same LM and evaluation dataset as in Section 3.1. Automatic evaluation results are summarized in Table 4, clearly showing an improvement over the baseline hybrid system (MHyT) that does not have a syntactic pre-processing step.

Table 4. Evaluation results: MHyT – baseline hybrid system, SyMHyT – syntax-based hybrid system.

System	BLEU		NIST	
	MHyT	SyMHyT	MHyT	SyMHyT
Google Translate	18.09		8.37	
Bing Translator	18.87		8.09	
LetsMT!	30.28		9.45	
Google + Bing	18.73	21.27	7.76	8.30
Google + LetsMT	24.50	26.24	9.60	9.09
LetsMT! + Bing	24.66	26.63	9.47	8.97
Google + Bing + LetsMT!	22.69	24.72	8.57	8.24

To evaluate the influence of language model size on the chunk selection process, we trained two 12-gram language models – one on the JRC corpus (Section 3.1) and another one on the DGT-Translation Memory (DGT-TM) corpus (Steinberger et al., 2012). The results of this experiment are presented in Table 5. The higher-order language model did not show improvement. Some additional experiments described in Section 3.3, using 6-gram, 9-gram and 12-gram LMs resulted in slightly higher BLEU score, but the change was not statistically significant.

Table 5. Influence of different language models.

LM	Size (sentences)	BLEU
5-gram JRC	1.4 million	**24.72**
12-gram JRC	1.4 million	24.70
12-gram DGT-TM	3.1 million	24.04

A random 2% (32 sentences) of the translations from the experiment were given to 10 native speakers of Latvian with instructions to evaluate fluency and adequacy. The three baseline systems were compared with the syntax-based hybrid system that combines all three baselines. Evaluators were instructed to mark each sentence with one of the following labels: "most fluent translation", "most precise translation", "neither most fluent, nor most precise", or "both most fluent and most precise". In case, if a translation is marked as most fluent and adequate, then all others alternatives needed to be marked as "neither most fluent nor most precise". Results of evaluation are summarized in Table 6. The free-marginal kappa (Randolph, 2005) for these annotations is 0.335 that indicates substantial agreement between the annotators.

Table 6. Manual evaluation results.

System	Fluency AVG	Accuracy AVG	SyMHyT selection	BLEU
Google	35.29%	34.93%	16.83%	18.09
Bing	23.53%	23.97%	17.94%	18.87
LetsMT	20.00%	21.92%	65.23%	30.28
SyMHyT	21.18%	19.18%	-	24.72

The table shows that about $\frac{1}{3}$ of translations recognized by annotators as most fluent and most adequate are translations from *Google Translate* system. This contradicts with the automatic evaluation results and the selections made by the syntax-based hybrid MT, where a tendency towards the LetsMT! translation is observed.

3.3. Combining sentence fragment translations - advanced fragmenting

This section presents several methods to enrich the MSMT system with linguistic knowledge. The experiments described use multiple combinations of outputs from two, three or four online MT systems. The approach allows to increase output by 1.48 BLEU points when translating general domain texts. The section is based on the paper of Rikters and Skadiņa (2016b).

The major components of the system are the same as in the previous section (3.2), and the general workflow is very similar to what was shown in Figure 2. When translation is performed into a morphologically rich language, a simple chunk translation approach may not lead to a better translation. For example, when small chunks are translated into Latvian, they usually will be in a canonical form that corresponds to the subject of the sentence but will be incorrect for the object. On the other hand, if long chunks are translated, then the translation usually breaks agreement rules, or the translation has an incorrect word order.

Experiments were conducted on the English – Latvian language pair. Two legal domain corpora – JRC and DGT-TM – were used for language modelling. For evaluation two different evaluation sets were used – 1) the evaluation set from Section

3.1; and 2) the ACCURAT[9] balanced test corpus (Skadiņa et al., 2010). As the baseline, we used full translations from each individual online API and simple MSMT system (Rikters 2015) that uses only perplexity to select the best translation from outputs of the online APIs.

We evaluated two approaches in chunk translation – translation of top-level chunks and translation of smaller chunks that are selected based on their properties in the sentence. In the first experiment (SyMHyT), a parse tree of each sentence is processed by the chunk extractor to obtain the top-level sub-trees (noun phrases, verb phrases, prepositional phrases, etc.). The chunk extractor uses regular expressions to identify sub-trees. When sub-trees are identified, they are translated with online APIs. Finally, the translation of the sentence is generated by a combination of translation hypothesis of sub-trees as it is described in section 3.3. We evaluated this approach for two SyMHyT systems: *Bing* + *Google* (BG) and *Bing* + *Google* + *Hugo* (BGH). An analysis of selected translated chunks revealed a discrepancy between BLEU score evaluation results and preferences of the selection module. In addition, we observed some apparent flaws, e.g. one-word chunks, one-symbol chunks or very long chunks. This motivated us to investigate more complex algorithm for chunk extraction.

The enhanced chunk extractor (ChunkMT) reads output from the Berkeley Parser and places it in a tree data structure. During this process, each node of the tree is initialised with its phrase (NP, VP, ADVP, etc.), word (if it has one) and a chunk consisting of the chunks from its child nodes. To obtain the final chunks for translation, the resulting tree is traversed bottom-up post-order (left to right). A chunk is combined with the previous one, if it is a) non-alphabetical, b) only one symbol, or c) contains a genitive phrase. If a chunk is very long (length of chunk > sentence length / 4 in the first chunking iteration), an attempt to break it into smaller chunks is made. Figure 3 illustrates chunk extraction result of both MSMT systems.

SyMHyT	**ChunkMT**
Recently there has been an increased interest in the automated discovery of equivalent expressions in different languages .	Recently there has been an increased interest in the automated discovery of equivalent expressions in different languages .

Figure 3. Examples of chunks extracted by SyMHyT and ChunkMT.

[9] ACCURAT balanced test corpus for under resourced languages: http://metashare.tilde.com/repository/browse/accurat-balanced-test-corpus-for-under-resourced-languages/09cf87927ef211e5aa3b001dd8b71c662b9642e71de848dd9e5c92c0ee97dd1d/

Analysis of selected chunks (

Table 7) revealed interesting phenomenon which needs further investigations – when all systems are combined, translations from the best baseline system is selected only in 33% of cases, but from the second-best system only in 16.59% of cases.

Table 7. Best results using evaluation data and LM from JRC and selected chunk percentages.

System	BLEU	Equal	Bing	Google	Hugo	Yandex
BLEU	-	-	16.99	16.19	20.27	19.75
MSMT - BG	16.38	4.88%	45.03%	50.09%	-	-
MSMT - BGH	17.89	2.78%	34.31%	28.93%	33.98%	-
SyMHyT - BG	17.36	4.59%	24.61%	70.80%	-	-
SyMHyT - BGH	19.50	2.88%	18.01%	15.71%	63.40%	-
ChunkMT - BG	17.67	15.23%	41.14%	43.63%	-	-
ChunkMT - HY	21.38	9.15%	-	-	44.79%	46.06%
ChunkMT - all	20.33	2.94%	27.80%	19.67%	33.00%	16.59%

For general domain data (Table 8), the best result (+1.48 BLEU) is obtained by combining output from all four MT systems. Just like for the legal domain, results of two system combination are better, when better baseline systems are combined. Increase by 0.56 BLEU points is observed when *Bing* and *Google* systems are combined (BG). Table 9 presents the distribution of selected translated chunks between different MT engines. Most of the translations are from *hugo.lv*, which can be explained with the choice of legal domain language model, while *Google* and *Bing* were the best baseline systems for the general domain.

Table 8. Evaluation results on ACCURAT balanced test corpus.

	12-gram		6-gram	
System	**BLEU**	**TER**	**BLEU**	**TER**
JRC LMs				
BG	17.34	0.757	17.30	0.757
HY	15.72	0.774	15.78	0.775
All	-	-	15.88	0.774
DGT LMs				
BG	**18.29**	0.753	17.81	0.760
HY	17.72	0.757	16.49	0.768
HG	18.06	**0.747**	-	-
All	**19.21**	**0.745**	16.36	0.776

Table 9. Best results using balanced evaluation data and DGT-TM LM and distribution of selected chunks.

System	BLEU	Equal	Bing	Google	Hugo	Yandex
BLEU	-	-	17.43	17.73	17.14	16.04
MSMT - BG	17.70	7.25%	43.85%	48.90%	-	-
MSMT - BGH	17.63	3.55%	33.71%	30.76%	31.98%	-
SyMHyT - BG	17.95	4.11%	19.46%	76.43%	-	-
SyMHyT - BGH	17.30	3.88%	15.23%	19.48%	61.41%	-
ChunkMT - BG	18.29	22.75%	39.10%	38.15%	-	-
ChunkMT - all	19.21	7.36%	30.01%	19.47%	32.25%	10.91%

3.4. Combining sentence fragment translations by exhaustively searching across possibilities

A problem with the approaches described in the previous sections is that they can potentially miss some certain combinations of chunks that only score a low perplexity when put together in a full sentence but not necessarily as individual chunks. With this in mind, as well as the increasing availability of high-performance software engineering techniques and computing resources for experimentation, it has become possible to not simply evaluate each individual translated chunk and combine them but also iterate through all variants of different combinations. Doing it this way allows for finding the best translation of a specific sentence that only 'looks' good as a whole but not necessarily that good as individual chunks. This section is based on the paper of Rikters (2016c).

The full search MT system combination approach (FuSCoMT) was developed based on ChunkMT (Section 3.3). Therefore, its architecture is very similar to ChunkMT but with few key differences. The workflow of the system can be decomposed into the following steps: pre-processing of the source sentence, acquisition of translations via online APIs, and generation of MT output, as it is shown in Figure 4. The main difference is in the last step - the manner of scoring chunks with the LM and selecting the best translation. The other significant change is the utilisation of multi-threaded computing that allows running the process on all available CPU cores in parallel.

As opposed to ChunkMT, it firstly generates all unique sequential combinations of translations, using the given chunks. The amount of the combinations is calculated as n^r where n is the number of different translation engines, and r is the number of chunks. Since the translation engines, in this case, are the same four as in ChunkMT, the combination count will be 4^r. The next step is the scoring of each full sentence perplexity, using the LM. Finally, when a perplexity score has been obtained for all full-sentence combinations, the lowest-scoring one is selected as the best candidate.

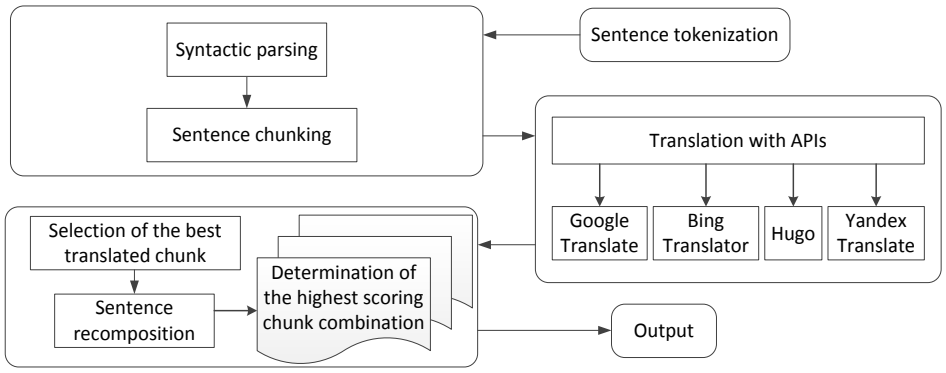


Figure 4. Workflow of the translation process.

As opposed to ChunkMT, it firstly generates all unique sequential combinations of translations, using the given chunks. The amount of the combinations is calculated as n^r where n is the number of different translation engines, and r is the number of chunks. Since the translation engines, in this case, are the same four as in ChunkMT, the combination count will be 4^r. The next step is the scoring of each full sentence perplexity, using the LM. Finally, when a perplexity score has been obtained for all full-sentence combinations, the lowest-scoring one is selected as the best candidate.

To make the experiments comparable to the baseline MSMT system, the same corpora were used for both – training the LM and preparing evaluation data. The automatic evaluation results of the experiments are shown in Table 10.

Table 10. Full search experiment results.

System		Full-search	ChunkMT	Bing	Google	Hugo	Yandex
BLEU	Legal	**23.61**	20.00	16.99	16.19	20.27	19.75
	General	14.40	17.27	17.43	17.72	17.13	16.03

3.5. Combining sentence fragment translations with neural network Language Models

This section presents an enrichment of the existing MSMT approach with the addition of neural language models. The core components of the system have not changed from the ones mentioned the previous sections. The section is based on the paper of Rikters (2016d).

The baseline LM was trained KenLM as in section 3.1. In order to outperform the baseline, 3 neural network (NN) LM toolkits were explored. The RWTHLM toolkit (Sundermeyer et al., 2014) has support for feed-forward, recurrent and long short-term

memory NNs (Hochreiter and Schmidhuber, 1997; Gers et al., 2000). The MemN2N toolkit allows training an end-to-end memory network (Sainbayar et al., 2015) LM. The final toolkit - Char-RNN[10] is a multi-layer recurrent NN for training character-level LMs.

For training the LMs the DGT-TM corpus was used (only the first half for Char-RNN). Evaluation and validation datasets were automatically derived from the training data with the proportion of 97%:1.5%:1.5%. Table 11 shows differences in perplexity evaluations that outline the superiority of NN LMs. It also shows that the statistical model is much faster to train on a CPU and that NN LMs train more efficiently on GPUs. The last column of Table 11 shows BLEU scores for different NN LMs. The results show that the approach that reaches the lowest LM perplexity slightly improves the final translations, as well as the LM with the highest perplexity leads to slightly worse translations. In these experiments, the LM quality impacts MT results in the range of 0.75 BLEU.

Table 11. Results of language model perplexity experiments.

System	Perplexity	Corpus size	Trained on	Training time	BLEU
KenLM	34.67	3.1M	CPU	**1 hour**	19.23
RWTHLM	136.47	3.1M	CPU	7 days	18.78
MemN2N	25.77	3.1M	GPU	4 days	18.81
Char-RNN	**24.46**	1.5M	GPU	2 days	**19.53**

4. Combining neural machine translation output

4.1. Finding correlation between neural network attention and output translation quality

NMT systems allow to save the attention values between input-output tokens. These values can be interpreted as the influence of the input token on the output token, or the strength of the connection between them. Thus, weak or dispersed connections should intuitively indicate a translation with low confidence, while high values and strong connections between one or two tokens on both sides should indicate higher confidence. Figure 6 shows an example of a translation that has little or nothing to do with the input, a frequent occurrence in NMT. This section is based on the paper of Rikters and Bojar (2017).

The experiments described in this section helped the author understand possible use-cases for NMT attention alignments, which were essential to enable NMT system combination described in sections 4.2 and 0. Multi-word expressions (MWEs) have been

[10] Multi-layer Recurrent Neural Networks (LSTM, GRU, RNN) for character-level language models in Torch https://github.com/karpathy/char-rnn

a challenge for SMT and NMT because they may not appear frequently enough in training data. In order to examine how MWEs are treated by NMT systems, we 1) trained baseline NMT systems; 2) extracted parallel MWE corpora from the training data; 3) trained NMT systems with synthetic MWE data; and 4) inspected attention alignments produced by the NMT.

Training and development corpora were used from the WMT 2017 shared tasks[11] (Bojar et al., 2017a). Neural Monkey (Helcl and Libovický, 2017), was used to train the NMT systems with configuration provided by the WMT Neural MT Training Task[12]. To extract MWEs, the corpora were first tagged with morphological taggers: UDPipe (Straka et al., 2016) for English (En) and Czech (Cs), LV Tagger (Paikens et al., 2013) for Latvian (Lv). After that, the tagged corpora were processed with the MWE toolkit (Ramisch, 2012), and finally aligned with the MPAligner (Pinnis, 2013). This workflow allowed to extract a parallel corpus of about 400 000 MWE candidates for English → Czech and about 60 000 for English → Latvian. Full sentences containing MWEs were also extracted from the training corpus, serving as a separate parallel corpus.

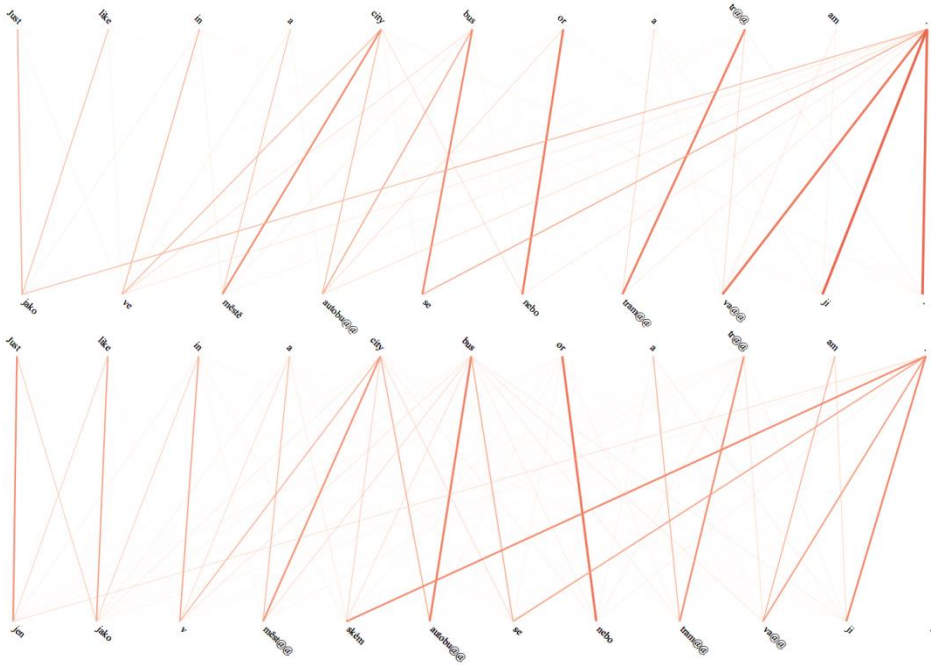
We experiment with two forms of the presentation of MWEs to the NMT system: 1) only parallel MWEs, and 2) full sentences containing MWEs. We denote the approaches *MWE phrases* and *MWE sents.* We mix the baseline parallel corpus with synthetic data so that MWEs get more exposure to the neural network in training and allow NMT to learn to translate them better. For En → Lv the full corpus was used. For En → Cs we used only the first 15M sentences. The MWEs were repeated five times in both language pairs. By doing this, the En → Cs dataset was reduced from 49M to 17M, and the En → Lv dataset increased to 4.8M parallel sentences for one epoch of training.

Table 12. BLEU scores of experiments.

Languages	En → Cs		En → Lv	
Dataset	Dev	MWE	Dev	MWE
Baseline	13.71	10.25	11.29	9.32
+MWE phrases	-	-	**11.94**	**10.31**
+MWE sents.	**13.99**	**10.44**	-	-

Table 12 shows the automatic evaluation results for each approach on one language pair. We evaluate all setups with BLEU on the full development set (distinct from the training set), as shown in the column *Dev*, and on a subset of 611 (En → Lv) and 112 (En → Cs) sentences containing the identified MWEs (column *MWE*).

[11] http://www.statmt.org/wmt17/translation-task.html
[12] http://www.statmt.org/wmt17/nmt-training-task



Source	Just like in a city bus or a tram.
Baseline	Jako ve městé autobuse nebo tramvaji.
Improved NMT	Jen jako v městském autobuse nebo tramvaji.
Reference	Stejně jako v městském autobuse či tramvaji.

Figure 5. Soft alignment example visualizations from translating an English sentence into Czech from the baseline (top, hypothesis 1) and improved (bottom, hypothesis 2) NMT systems.

For inspecting the NMT attention alignments, we developed a tool (Rikters et al., 2017a) that takes data produced by Neural Monkey as input and produces a soft alignment visualization by connecting words and subword units (Sennrich et al., 2016b) as shown in Figure 5, which shows an example translation with two systems for En → Cs. Here it is clear that in the baseline alignment no attention goes to the word "městě" or the subword units "autobu@@" and "se" when translating "city". In the modified version, on the other hand, some attention from "city" goes into all closely related subword units: "měst@@", "ském", "autobu@@", and "se". It is also shown that in this example, the translation of "bus" gets attention from not only "autobu@@" and "se", but also the ending subword unit of "city", i.e. the token "ském".

4.2. Simple system combination using neural network attention

This section describes NMT systems built by the combined effort of the University of Latvia, University of Zurich and University of Tartu. We participated in the WMT 2017 shared task on news translation by building systems for two language pairs: English ↔ German and English ↔ Latvian. We trained several baseline systems with two NMT and one SMT framework - Nematus (NT), Neural Monkey (NM) and LetsMT! (LMT). To outperform the baselines, we explored 4 areas for improvements – 1) filtering back-translated data; 2) named entity forcing; 3) hybrid system combination; and 4) NMT-specific post-processing. The section is based on the paper of Rikters et al. (2017a).

We used each of our NMT systems to back-translate 4.5 million sentences of news corpora in each translation direction. We trained an LM using Char-RNN with 4 million sentence news corpora of the target languages, resulting in three character-level LMs - English, German (De) and Latvian. We used them to get perplexity scores for translations, ordered them by perplexity and used the top 50% together with the sources as sources and references respectively as additional filtered synthetic in-domain corpora.

For English ↔ German, we enforced the translation of named entities (NE) using a custom dictionary. We performed named entity recognition (NER) using spaCy[13] for German and NLTK[14] for English, aligned the recognised entities with GIZA++ (Och and Ney, 2003), and created an entry in our dictionary for every pair of aligned NEs. We filtered the dictionary by automatically removing entries that: 1) did not contain alphabetical characters; 2) were longer than 70 characters or five tokens; 3) were differed from each other in length by more than 15 characters or two tokens; 4) started with a dash. During translation we identified NEs in the source text, for every NE, we checked whether there was a translation in our dictionary and swapped the identified aligned translation with the one from the dictionary. If it was not in the dictionary, we copied the verbatim NE expression from the source sentence to the target sentence.

For translating between English ↔ Latvian, we used all 3 systems in each direction and obtained the attention alignments from the NMT systems. For each direction, we chose one main NMT system to provide the final translation for each sentence and, judging by the attention alignment distribution, tried to automatically identify unsuccessful translations. Two main types of unsuccessful translations that we noticed were: 1) when the majority of alignments are connected to only one token (example in Figure 6), or, 2) when all tokens strongly align one-to-one, suggesting that the source may not have been translated at all (example in Figure 11). In the case of an unsuccessful translation, the hybrid setup checks the attention alignment distribution from the second NMT system and outputs either the sentence of that or performs a final back-off to the SMT output.

[13] Industrial-Strength Natural Language Processing in Python - https://spacy.io
[14] Natural Language Toolkit - http://www.nltk.org

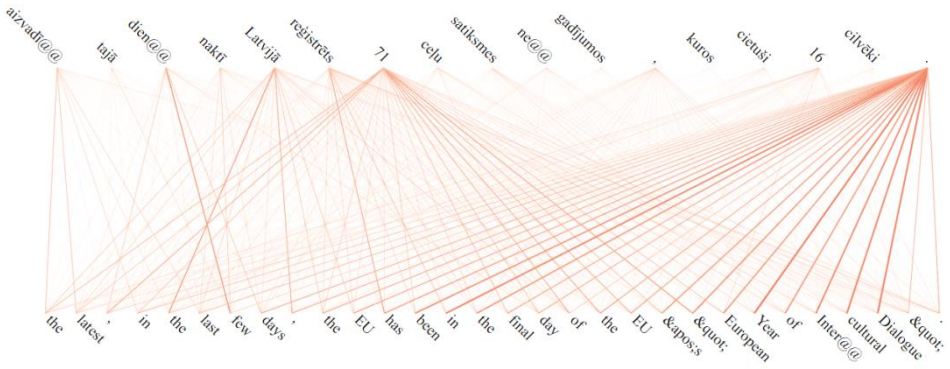


Figure 6. Attention alignment visualization for a bad translation. Reference translation: 71 traffic accidents in which 16 persons were injured have happened in Latvia during the last 24 hours., hypothesis translation: the latest , in the last few days , the EU has been in the final day of the EU 's " European Year of Intercultural Dialogue ". Confidence scores (details in section 0): CDP = -0.900, AP_{out} = -2.809, AP_{in} = -2.137, Total = -5.846.

In post-processing of translation output, we aimed to fix the most common mistakes that NMT systems tend to make. We used the output attention alignments from the NMT systems to replace <*unk*> tokens with the source tokens that are aligned to them with the highest weight. Any consecutive repeating n-grams were replaced with a single n-gram. The same was applied to repeating n-grams that have a preposition between them, e.g., *victim of the victim*.

Table 13. Experiment results for translation between English ↔ German on development (newsdev2017) and evaluation (newstest2017) sets. Submitted systems are in bold.

System	En → De		De → En	
Dataset	Devel.	Eval.	Devel.	Eval.
Baseline NT	27.4	21.0	31.9	27.2
+Filtered synthetic data	30.7	22.5	36.8	28.8
+NE forcing	30.9	**22.7**	36.9	**29.0**

Table 14. Experiment results for translating between English ↔ Latvian on development (*newsdev2017*) and evaluation (*newstest2017*) sets. Submitted systems are in bold.

System	En → Lv		Lv → En	
Dataset	Devel.	Eval.	Devel.	Eval.
Baseline NM	11.9	11.9	14.6	12.8
Baseline NT	12.2	10.8	13.2	11.6
Baseline LMT	19.8	12.9	24.3	13.4
NM +filtered synthetic data	16.7	13.5	15.7	14.3
NT +filtered synthetic data	16.9	13.6	15.0	13.8
NM+NT+LMT	-	**13.6**	-	**14.3**

The final results of our English ↔ German systems are summarized in Table 13 and the results of our English ↔ Latvian systems - in Table 14. Table 15 shows how our systems were ranked in the WMT17 shared news translation task against other submitted primary systems in the constrained track (Bojar et al., 2017b). Since the human evaluation was performed by showing evaluators only the reference translation and not the source, the human evaluation rankings are the same as BLEU, which also considers only the reference translation. One exception is the ranking for En ↔ Lv, where an insufficient amount of evaluations was performed to cover all submitted systems, resulting in a tie for the 1st place across all but one submitted systems.

Table 15. Automatic (BLEU) and human ranking of our submitted systems (C-3MA) at the WMT17 shared news translation task, only considering primary constrained systems. Human rankings are shown by clusters according to Wilcoxon signed-rank test at p-level p<=0.05, and standardized mean DA score (Ave %).

System	Rank		
	BLEU	Human	
		Cluster	Ave %
De → En	6 of 7	6-7 of 7	7 of 7
En → De	10 of 11	9-11 of 11	9 of 11
En → Lv	11 of 12	1-11 of 12	11 of 12
Lv → En	5 of 6	4-5 of 6	4 of 6

4.3. System combination by estimating confidence from neural network attention

This section proposes usage of NMT attention alignments as an indicator of the translation output quality and the confidence of the decoder. It is based on the paper of Rikters and Fishel (2017).

Besides the text of the translation that was shown in Figure 6, it is clear already by looking at the attention weights of this pair that the translation is weak, due to many input tokens (like the sentence-final full-stop) being most strongly connected to several unrelated output tokens. In other words, their coverage is too high. We introduce several metrics to formalize this intuition: penalizing translations with tokens with a total coverage of not just below but much higher than 1.0, as well as tokens with a dispersed attention distribution.

The first part of our metric draws inspiration from the coverage penalty (Wu et al., 2016b); however, it penalizes not just lacking attention but also too much attention per input token. The aim is to penalize the sum of attentions per input token for going too far from 1.0, so tokens with total attention of 1.0 should get a score of 0.0 on the logarithmic scale, while tokens with less attention (like 0.2) or more attention (like 2.5) should get lower values. We thus define the coverage deviation penalty (5),

where L is the length of the input sentence, i is the output token index, j - the input token index, α – attention probability. The metric is on a logarithmic scale, and it is normalised

by the length of the input sentence in order to avoid assigning higher scores to shorter sentences.

$$CDP = \frac{1}{L}\sum_j \log\left(1 + \left(\sum_i \alpha_{ji}\right)^2\right) \quad (5)$$

It is not enough to simply cover the input; we conjecture that more confident output tokens will allocate most of their attention probability mass to one or a small number of input tokens. Thus, the second part of our metric is called the absentmindedness penalty (6), and targets scattered attention per output token, where the dispersion is evaluated via the entropy of the predicted attention distribution. Again, we want the penalty value to be 1.0 for the lowest entropy and head towards 0.0 for higher entropies. The values are again on the log-scale and normalised by the source sentence length L (i is the output token index, j - the input token index, α – attention probability).

$$AP_{out} = -\frac{1}{L}\sum_i \sum_j \alpha_{ji} \cdot log\alpha_{ji} \quad (6)$$

The absentmindedness penalty can also be applied to the input tokens after normalising the distribution of attention per input token, resulting in the counter-part metric AP_{in}. This is based on the assumption that it is not enough to cover the input token, but rather the input token should be used to produce a small number of outputs. Finally, we combine the coverage deviation penalty with both the input and output absentmindedness penalties into a joint metric via summation (7).

$$confidence = CDP + AP_{out} + AP_{in} \quad (7)$$

To evaluate the metrics, we applied them to filter translations and incorporated them into a sentence-level hybrid translation scheme. We trained baseline systems with two NMT frameworks - Nematus (NT) and Neural Monkey (NM). For the baseline systems, we used all available parallel data from the WMT17 news translation task[15] for En ↔ De and En ↔ Lv.

We used our baseline En → Lv and Lv → En NM and NT systems to translate all available Latvian monolingual news domain data - 6.3 million sentences in total, and the first 6 million sentences from the English *News Crawl 2016*. For each translation, we used the attention provided from the NMT system to calculate our confidence score, sorted all translations according to the score and selected the top half of the translations along with the corresponding source sentences as the synthetic parallel corpus. For comparison, we trained a Char-RNN LM with 4 million sentences from news domain for each of the target languages and used them to get perplexity scores for all translations, order them and get the *better half*.

[15] EMNLP 2017 Second Conference on Machine Translation - http://www.statmt.org/wmt17

Table 16. Experiment results in BLEU for translating between English ↔ Latvian with different types of back-translated data using development (200 random sentences from *newsdev2017*) and evaluation (*newstest2017*) datasets.

	BLEU			
System	**En → Lv**		**Lv → En**	
Dataset	**Devel.**	**Eval.**	**Devel.**	**Eval.**
Baseline NM	8.36	11.90	8.64	12.40
NM + Full Synthetic	9.42	13.50	9.01	13.81
NM + LM-Filtered Synthetic	9.75	13.52	9.45	14.30
NM + Attention-Filtered Synthetic	8.99	12.76	11.23	14.83

We shuffled each synthetic parallel corpus with the baseline parallel corpora and used them to train NMT systems. We also trained a system with the full set of back-translated data for each translation direction. Results on a subset of *newsdev2017* and the full *newstest2017* dataset are summarized in

Table **16**. As expected, adding back-translated synthetic training data allows to get higher BLEU scores in all cases. It can be observed that filtering out half of the poorly translated data and keeping only the best translations either does not decrease the final output quality in some cases or even further increase the quality in others when using the LM.

We translated the development set with both baseline systems for each language pair in each direction. We used the confidence score to compare both translations of a source sentence and choose the better one. Results of the hybrid selection experiments are summarized in Table 17. For translating between En ↔ Lv, where the difference between the baseline systems is not that high (0.06 and 1.55 BLEU), the hybrid method achieves some meaningful improvements. However, for En ↔ De, where differences between the baseline systems are more significant (3.46 and 4.46 BLEU), the hybrid drags both scores down.

Table 17. Hybrid selection experiment results in BLEU on the development dataset (200 random sentences from *newsdev2017*).

System	**En → De**	**De → En**	**En → Lv**	**Lv → En**
Neural Monkey	18.89	26.07	13.74	11.09
Nematus	**22.35**	**30.53**	13.80	12.64
Hybrid	20.19	27.06	**14.79**	**12.65**
Human	23.86	34.26	15.12	13.24

The last row in Table 17 shows BLEU scores for the scenario when human annotator preferences were used to select output sentences. An overview of human evaluator preferred translation selections is summarised in Table 18. The results show that out of all translations the human evaluators deliberately prefer one or the other system. Aside from En - Lv, where a slight tendency towards Neural Monkey translations can be observed, all others look more or less equal. This highly contrasts with the BLEU scores

from Table 17, where for both translation directions from English human evaluators prefer the lower-scoring system more often than the higher-scoring one. The final row of Table 18 shows how much our attention-based score matches human judgments in selecting the best translation.

Table 18. Human evaluation results on 200 random sentences from the newsdev2017 dataset.

System	En → De	De → En	En → Lv	Lv → En
Neural Monkey	54%	42%	61.5%	47%
Nematus	46%	58%	38.5%	53%
Overlaps with hybrid	**57%**	47%	**62.5%**	51%

4.4. Data combination for training multilingual neural machine translation systems

This section describes experiments in combining data from 3 different languages to train a single NMT model for translating from and into multiple languages. 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 modifications to the data iterator module for it to automatically read the multi-way training data in equal batches for each translation direction and add the target language symbol at the beginning of each source sentence. The section is based on the publications of Rikters et al. (2018a) and Rikters et al. (2018b).

We trained multiplicative long short-term memory (Krause et al., 2017) (MLSTM-SO – the baseline) models and gated recurrent units (GRU-SM, GRU-DO and GRU-DM) models with Nematus (Sennrich et al., 2017), fully convolutional neural network models - (FConv-O and FConv-M) and transformer neural network models (Transf.-O and Transf.-M) with Sockeye (Hieber et al., 2017). The model and data configurations were either shallow (S) or deep (D) in the case of GRU and either one-way (O) or multi-way (M).

We used En ↔ Ru, En ↔ Et, and Ru ↔ Et data of multiple publicly available and proprietary datasets for training. One-way models were trained on En ↔ Et and Ru ↔ Et data and multi-way models were trained on data from all language pairs in both directions. The corpora were cleaned and filtered using scripts from Pinnis et al. (2017). An overview of the training data statistics before and after filtering for each language pair is given in Table 19.

Table 19. Training data sentence counts before and after filtering.

Language pair	Before filtering (Total/Unique)	After filtering (Unique)
English ↔ Estonian	62.5M / 24.3M	18.9M
English ↔ Russian	60.7M / 39.2M	29.4M
Russian ↔ Estonian	6.5M / 4.4M	3.5M

For Estonian ↔ Russian, we selected 2000 random sentences from the training data to be used as development data. Development datasets for all other translation directions were obtained from the ACCURAT development datasets (Skadiņa et al., 2010). In the multi-way model training scenarios, we concatenated 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, for which the Russian version was prepared manually.

We were mainly focused on improving the translation quality when translating between Russian (Ru) and Estonian (Et) because this specific language pair had the weakest performance among the baseline systems. Table 20 shows how each of the models compares to the baseline.

Table 20. Translation automatic evaluation results for all model architectures on development and evaluation data. The best results are in bold.

	Development				Evaluation			
	Ru → Et	Et → Ru	En → Et	Et → En	Ru → Et	Et → Ru	En → Et	Et → En
MLSTM-SO	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-DO	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-O	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
Transf.-O	17.44	18.90	**25.27**	**37.12**	9.10	11.17	**28.43**	**40.08**
Transf.-M	**18.03**	**19.18**	23.99	35.15	**14.38**	**15.48**	25.56	37.97

The results show that the deep GRU multi-way model outperforms the one-way models in most cases. However, the convolutional and transformer models increase quality only for the low-resource language pairs. The quality improvement for Et ↔ Ru ranges from 2.16 BLEU points (for FConv-M on the Et → Ru evaluation set) up to 5.28 BLEU points (for Transf.-M on the Ru → Et evaluation set). For the high-resource language pairs, both FConv-M and Transf.-M models show significantly lower translation quality than their respective one-way models. The quality decrease ranges from -2.11 BLEU points (for Transf.-M on the Et → En evaluation set) down to -5.17 BLEU points (for FConv-M on the Et → En evaluation set). This shows that the newer NMT architectures in multi-way scenarios are beneficial only to low-resource language pairs. 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.

5. Practical implementations

5.1. Interactive multi-system machine translation

This section describes a system for interactive MSMT that uses syntactic and statistical features and visualizes intermediate steps. Components from Section 3.3 were used as the back-end system (workflow visualized in Figure 2). The system allows either to combine output from online MT systems or user input translations (Figure 7). Afterwards, the system will perform syntactic analysis on the input sentence and split it into chunks as shown in Figure 8. The section is based on the paper of Rikters (2016a).

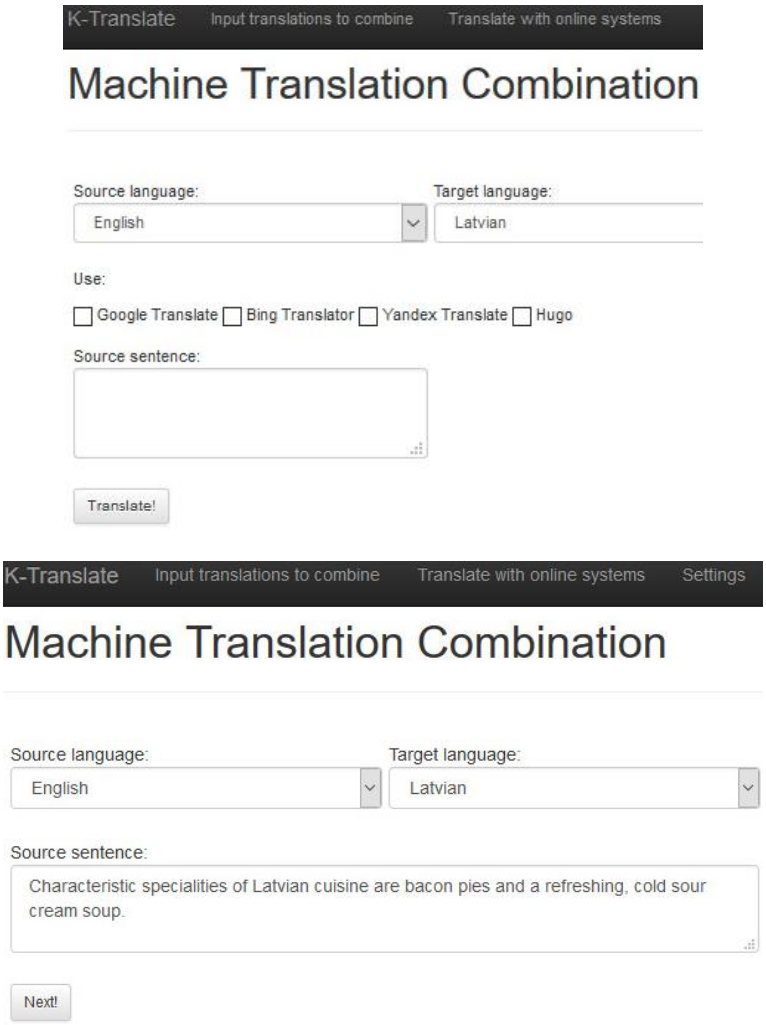


Figure 7. First step - translating with online APIs (top) or combining multiple user provided translations (bottom).

In the final step (

Figure **9**) the system provides the best combined translation and highlights which chunks were used from which input. It also shows the source used for each chunk and the confidence level of each selection. The confidence is calculated by comparing chunk perplexities to each other.

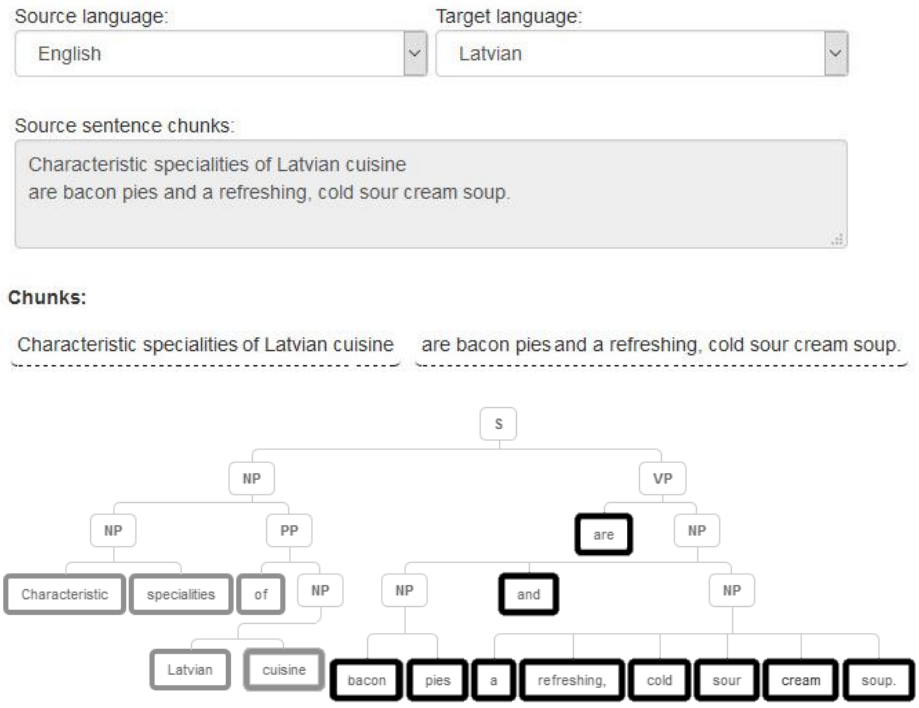


Figure 8. Second step – input sentence chunking and syntax tree visualization.

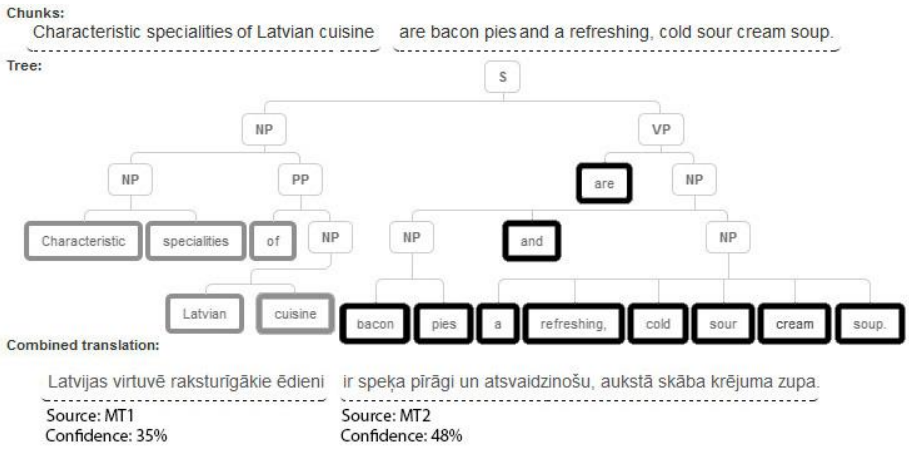


Figure 9. Final step - translation combination results page.

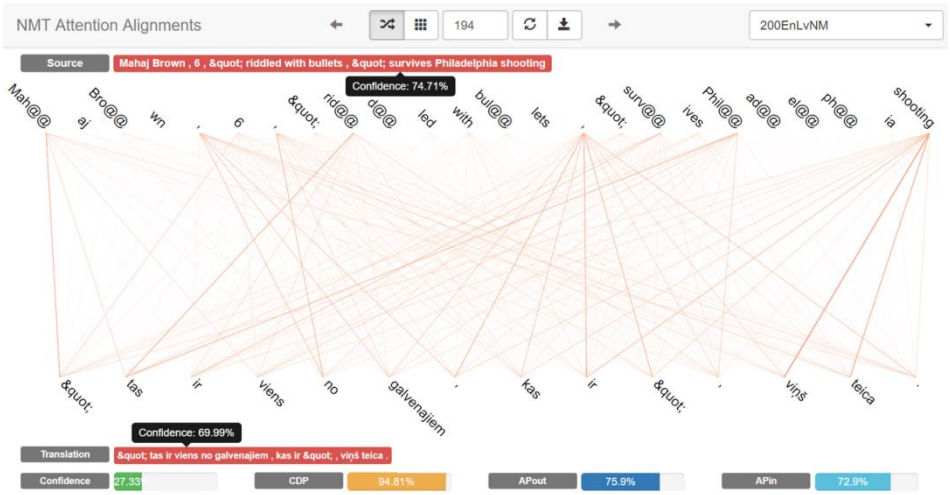
5.2. Visualizing and debugging neural machine translations

This section introduces a translation inspection tool that explicitly targets NMT output, using attention weights corresponding to specific token pairs during the decoding process, by turning them into visual representations that can help humans better understand how output translations were produced. A key difference from other similar tools is that to distinguish acceptable outputs from completely unreliable ones no reference translations are required. The section is based on the papers of Rikters et al. (2017b) and Rikters (2018a).

The web visualization is intended to provide an intuitive overview of one or multiple translated test sets by showing one sentence at a time with navigation to other sentences by ID, length or multiple confidence measures (section 0). For each individual sentence, four confidence metrics are shown, and a confidence score for each source and translated token (or subword unit). The alignment is represented in the following way: source tokens (at the top) are connected to translated tokens (at the bottom) via orange lines, ranging from entirely faint to very thick, as shown in Figure 10 and Figure 11. A thicker line from a translated token to a source token means that the decoder paid more attention to that source token when generating the translation. Ideally, these lines should mostly be thick with some thinner ones in between. When they look chaotic, connecting everything to everything (Figure 10) or everything in the translation to mostly just one token in the source, that can be an indication of an unsuccessful translation that will possibly have no relation with the source sentence. On the other hand, if all lines are thick, straight downwards, connected one-to-one (right part of Figure 11), that may point to nothing being translated at all. Additionally, the matrix style visualization is also available in the web version as shown on the left part of Figure 11.

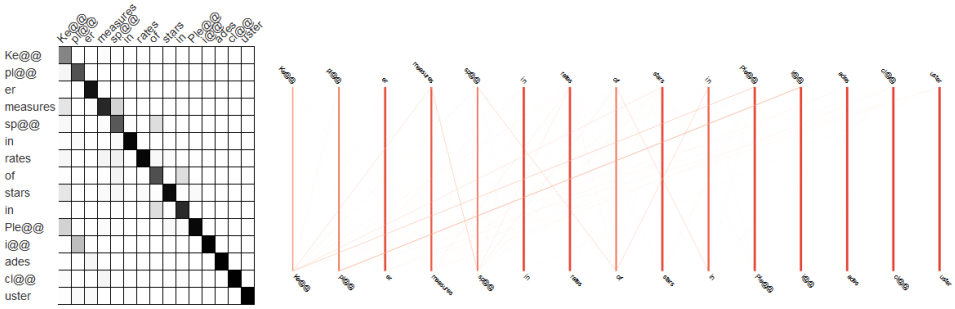
For each sentence, the tool displays an overall confidence score, coverage deviation penalty, and input and output absentmindedness penalties. The overall confidence score is also shown for each source token, indicating the amount of confidence that the token has been used to generate a correct translation, as well as for each translated token, indicating the amount of confidence that it is a correct translation. All of these scores are represented on a scale from 0 to 100 and can be used to navigate through the test set (Figure 12).

The confidence score considers hypotheses translations that are long and have a significant overlap with the source sentence as worse translations while tolerating considerable overlap for shorter sentences. The overlap ratio also serves as an individual score for sorting, navigating and comparing sentences from a dataset as shown in Figure 13.



Source	Mahaj Brown , 6 , "riddled with bullets ," survives Philadelphia shooting
Hypothesis	"tas ir viens no galvenajiem , kas ir" , viņš teica.
Reference	6 gadus vecais Mahajs Brauns "ložu sacaurumots" izdzīvo apšaudē Filadelfijā.

Figure 10. An example of a translated sentence that exhibits a low confidence score. Confidence: **27.33%**; CDP: 94.81%; AP_{out}: 75.9%; AP_{in}: 72.9%.



Source	Kepler measures spin rates of stars in Pleiades cluster
Hypothesis	Kepler measures spin rates of stars in Pleiades cluster
Reference	Keplers izmēra zvaigžņu griešanās ātrumu Plejādes zvaigznājā.

Figure 11. An example of a translated sentence that exhibits a suspiciously high confidence score. The translation here is a verbatim rendition of the input. Matrix form visualization on the left, line form visualization on the right. Confidence: **95.44%**; CDP: 100.0%; AP_{out}: 98.84%; AP_{in}: 98.85%.

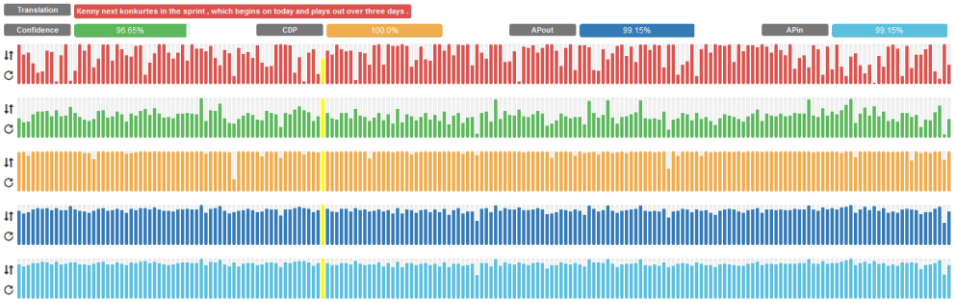
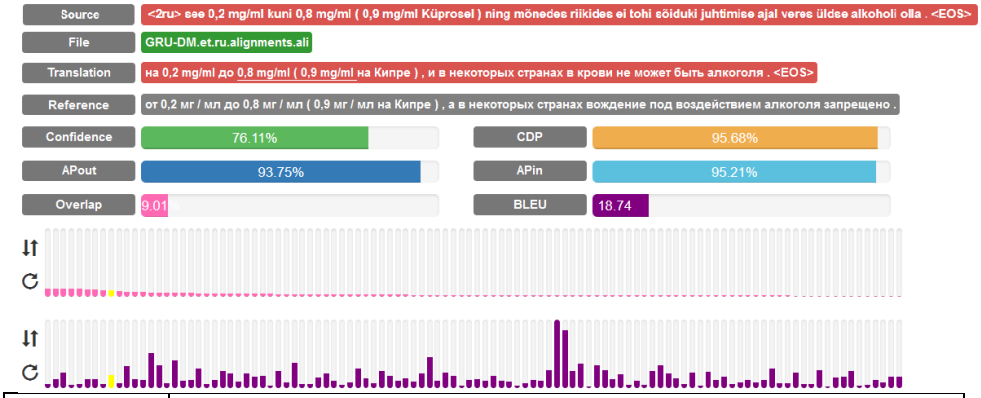


Figure 12. Navigation charts allow to jump to a sentence based on its length in characters (red), confidence (green), coverage deviation penalty (dark yellow), absentmindedness penalty for input (dark blue) and output (light blue). The currently active sentence is highlighted in bright yellow. All charts are sortable and scrollable for better user experience.

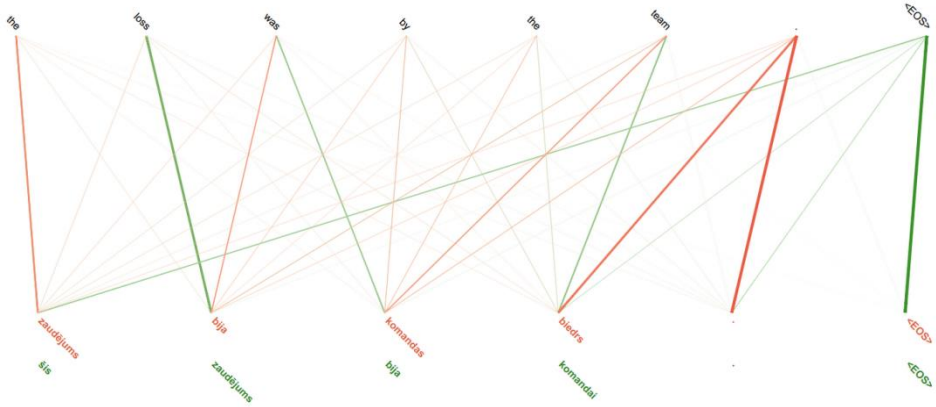


Source	see 0,2 mg/ml kuni 0,8 mg/ml (0,9 mg/ml Küprosel) ning mõnedes riikides ei tohi sõiduki juhtimise ajal veres üldse alkoholi olla.
Hypothesis	на 0,2 mg/ml до 0,8 mg/ml (0,9 mg/ml на Кипре) , и в некоторых странах в крови не может быть алкоголя.
Match	0,8 mg/ml (0,9 mg/ml

Figure 13. An example translation from Estonian into Russian, showing useful features for debugging translation outcomes - underlining of the longest matching substring between the source and translated sentences; sorting translations by overlap (pink bars) or BLEU score (purple bars); reference translation (grey background).

The tool has an option to compare two translations of the same source sentence directly. To perform the comparison, all source sentences for both input datasets must match, but the target sentences may differ in output token order as well as count. Comparisons may be performed between translations obtained from any two of the five supported NMT frameworks (Nematus, Neural Monkey, OpenNMT (Klein et al., 2017), Marian (Junczys-Dowmunt et al., 2016) and Sockeye (Hieber et al., 2017)). Figure 14 shows an example comparison of a sentence translated by two different NMT systems. On the top row is the source text and the bottom rows represent output from each

individual NMT system colour-coded to match the colours of the alignment lines. The second hypothesis (in green) exhibits stronger and more reliable output alignments to the content words while the first shows strong alignments coming from the stop sign. In this example, neither hypothesis matches the reference, but since it is only two words long for a source sentence of triple the length, it can hint to an oversimplified translation by the translator (assuming English was the original) and does not mean that both hypotheses are completely wrong. In fact, the second hypothesis is a fairly decent representation of the source sentence.



Source	the loss was by the team.
Hypothesis 1	zaudējums bija komandas biedrs.
Hypothesis 2	šis zaudējums bija komandai.
Reference	zaudē komanda.

Figure 14. A direct comparison of attention alignments of translating the same sentence with two different NMT systems.

A stronger penalty (8) is allocated to longer sentences that copy large amounts from the source, while shorter ones get more tolerance (e.g., the three-word English sentence "Thanks Barack Obama." can be perfectly translated into "Paldies Barack Obama." although $\frac{2}{3}$ of words in the translation are the same in the source). L_t - length of the target sentence; S - similarity between the source sentence and the translation on the scale of 0 - 1.

$$OP = (0.8 + (L_t * 0.01)) * (3 - ((1 - S) * 5)) * (0.7 + S) * tan(S)$$

The final confidence score sums up all three above mentioned metrics (9). For visualization purposes, each of the scores needed to be set on the same scale of 0-100%. To achieve that, we applied (10), where *X* is the score to convert and *C* is a constant of either 1 for the *CDP* or 0.05 for the other scores (AP_{out}, AP_{in}, *confidence*).

$$confidence = \begin{cases} CDP + AP_{out} + AP_{in},\ if\ similarity < 0.3 \\ CDP + AP_{out} + AP_{in} - OP,\ otherwise \end{cases} \quad (9)$$

$$\text{percentage} = e^{-C(X^2)} \quad (10)$$

5.3. Cleaning corpora to improve neural machine translation performance

This section describes how defective data affects NMT system training, as well as how to identify and deal with such data. We also show that using data filters in a hybrid multi-pass NMT setup can achieve state-of-the-art MT performance. The section is based on the papers of Rikters (2018b) and Pinnis et al. (2018).

A common defect in parallel corpora is a high mismatch of the number of non-alphabetic characters between source and target sentences (Figure 17). Often there are sentences that are entirely or mostly composed of characters outside the scope of the language in question (Figure 15). Another problem in parallel corpora is the same sentence of one language aligned to multiple different ones of the other language (Figure 16), but this is not always a bad indication, since they may just be paraphrases of the same concept (Figure 18). It is also wise to check if sentences in specific languages actually consist of text in that language (Figure 19) as there may be citations and other parts of foreign language texts, especially in news domain corpora. In automatically generated (automatically translated) parallel corpora multiple repetitions of tokens can sometimes be found (Figure 20). Sentences like this may not always be incorrect, but they introduce ambiguity when used to train MT systems.

Ô²Õ¡Ö Õ¥Ö Õ Õ¥Õ¦,	ლÇáÝÇÑÓ ÇáäÈíáლ,Nader-87	□□□□□□□□□□□□□ ".
MÃ©szÃ¡ros IstvÃ¡n	الـ فلسفة	走った森,

Figure 15. Examples of sentences with over 50% non-alphabetical symbols.

English	Estonian
I voted in favour.	kirjalikult. – (IT) Hääletasin poolt.
I voted in favour.	Ma andsin oma poolthääle.

Figure 16. An example of an English sentence aligned to multiple different Estonian sentences.

English	Estonian
Add to my wishlist	Hommikul (200 + 200 = 400 kcal)
Dec 2009	ÊßÇÌí 2009

Figure 17. An example of a high mismatch in non-alphabetical character counts between source and target.

English	Estonian
That is the wrong way to go.	See ei ole õge.
This is simply wrong.	See ei ole õge.

Figure 18. Multiple English paraphrased sentences aligned to one Estonian sentence.

English	Estonian
Zaghachi See okwu	3 Comments
Täna mängitud: 25 910	Täna mängitud: 25 929

Figure 19. Examples of sentences with a different identified language than the one specified.

English	Estonian
1 If , , and are the roots of , compute .	1 Juhul kui , Ja on juured , Arvutama .
we have that and or or or .	meil on, et ja või või või .
NXT Spray - NAPURA	NXT SPRAY NXT SPRAY

Figure 20. An example of repeating tokens (underlined).

To tackle the mentioned problems in parallel corpora, we introduce several filters:

- **Unique parallel sentence filter** – removes duplicate source-target sentence pairs.
- **Equal source-target filter** - removes sentences that are identical in the source side and the target side of the corpus.
- **Multiple sources - one target** and **multiple targets - one source** filters – remove repeating sentence pairs where the same source sentence is aligned to multiple different target sentences and multiple source sentences aligned to the same target sentence.
- **Non-alphabetical filters** – remove sentences that contain > 50% non-alphabetical symbols on the source or the target side, and sentence pairs that have significantly more (at least 1:3) non-alphabetical symbols in the source side than in the target side (or vice versa).
- **Repeating token filter** – removes sentences with consecutive repeating tokens or phrases.
- **Correct language filter** –estimates the language of each sentence (Lui and Baldwin, 2012) and removes any sentence that has a different identified language from the one specified.

We used the filters to clean parallel corpora provided in the WMT17[16] and WMT18[17] news MT shared tasks for English ↔ Estonian / Finnish (Fi) / Latvian. Detailed results of the cleaning process for three of the largest corpora - ParaCrawl, Rapid corpus of EU press releases (Rapid) and European Parliament Proceedings Parallel Corpus (Europarl) - are shown in Table 21. The results show that ParaCrawl is the most problematic corpus,

[16] Second Conference on Machine Translation - http://statmt.org/wmt17
[17] Third Conference on Machine Translation - http://statmt.org/wmt18

especially the Estonian part, where 85% had to be removed. The Rapid corpus had an overall higher quality with about 25% of parallel sentences removed. Europarl was by far the cleanest corpus, having only 5-6% of sentences removed by the cleaning toolkit.

We combined and shuffled all three En-Et corpora, resulting in 1 012 824 (46.50% of total) sentence parallel corpus for training NMT systems. The total amount of En-Fi parallel sentences was 2 719 104 (82.72% of total) after adding a cleaned version of the Wiki Headlines corpus, and En-Lv - 1 617 793 (35.85% of total) parallel sentences after adding cleaned versions of LETA translated news, Digital Corpus of European Parliament (DCEP), and Online Books corpora. We used the development datasets provided by the WMT shared tasks.

To observe the benefit of filtering data for NMT, we trained NMT models using filtered and non-filtered data in both translation directions for the three language pairs. We used Sockeye to train transformer architecture models until convergence on development data.

Table 21. Detailed results on filtering English-Estonian/Finnish/Latvian larger common parallel corpora from WMT shared tasks.

	Paracrawl		**Rapid**			**Europarl**		
	En Et	**En Fi**	**En Et**	**En Fi**	**En Lv**	**En Et**	**En Fi**	**En Lv**
Corpus size	1298103	624058	226978	583223	306588	652944	1926114	638789
Unique	26	37	23	161463	80894	23218	52686	19652
	0.00%	0.01%	0.01%	27.68%	26.39%	3.56%	2.74%	3.08%
src == tgt	242816	41611	428	3488	2929	490	528	707
	18.71%	6.67%	0.19%	0.60%	0.96%	0.08%	0.03%	0.11%
* sources 1 target	267235	17239	1108	1513	990	1176	6631	979
	20.59%	2.76%	0.49%	0.26%	0.32%	0.18%	0.34%	0.15%
* targets 1 source	69225	9532	752	1016	329	462	3536	435
	5.33%	1.53%	0.33%	0.17%	0.11%	0.07%	0.18%	0.07%
> 50% non-alpha	200338	12919	1226	5647	1699	66	285	72
	15.43%	2.07%	0.54%	0.97%	0.55%	0.01%	0.01%	0.01%
Non-alpha mismatch	23777	12737	6674	13311	6361	7211	24847	4012
	1.83%	2.04%	2.94%	2.28%	2.07%	1.10%	1.29%	0.63%
Repeating tokens	11210	1397	175	396	171	727	2594	703
	0.86%	0.22%	0.08%	0.07%	0.06%	0.11%	0.13%	0.11%
Language mismatch	283152	36233	14762	24854	8739	8924	10932	3301
	21.81%	5.81%	6.50%	4.26%	2.85%	1.37%	0.57%	0.52%
Total removed	1097779	131705	25148	211688	102112	42274	102039	29861
	85%	**21%**	**11%**	**36%**	**33%**	6%	5%	5%

The final NMT system results in Table 22 show that corpora filtering improves NMT quality for Estonian and Latvian systems, but not Finnish. The lack of improvement for Finnish is mainly due to the Europarl being the largest and at the same time the cleanest corpus for this language pair. The biggest corpora for Estonian and Latvian - ParaCrawl and DCEP respectively, were also the most problematic ones with 85% and 78% sentences removed respectively.

Table 22. Translation automatic evaluation results (BLEU scores) for all translation directions on development data. The best results are marked in bold. The second row shows how much of the initial parallel corpora remained after filtering for each language pair.

	En-Et	Et-En	En-Fi	Fi-En	En-Lv	Lv-En
Unfiltered	15.45	21.55	**20.07**	**25.25**	21.29	24.12
Corpus after filtering	46.50%		82.72%		35.85%	
Filtered	**15.8**	**21.62**	19.64	25.04	**22.89**	**24.37**
Difference	**+0.35**	**+0.07**	-0.43	-0.21	**+1.60**	**+0.25**

To test the full potential of the described methods, the highest-scoring En ↔ Et and En ↔ Fi models were further developed and submitted to the WMT 18 shared task: machine translation of news. The submitted systems were named *tilde-c-nmt-2bt* and *tilde-c-nmt-1bt* respectively. All developed systems ranked in the top 3-7 by automatic evaluation (BLEU score) out of 17-23 submissions in the constrained track (using only resources provided in the shared task).

To get the highest-quality translation results, we used a multi-pass hybrid approach for training NMT systems. With each trained NMT system, we supplemented the parallel training data with an additional set of back-translated (BT) for the next system (see Figure 21) resulting in multiple passes of training data during training. The final translations are produced using only the final NMT system (i.e., NMT3), unlike the multi-pass approach mentioned in Section 2.4, in which each input sentence is passed through multiple MT systems.

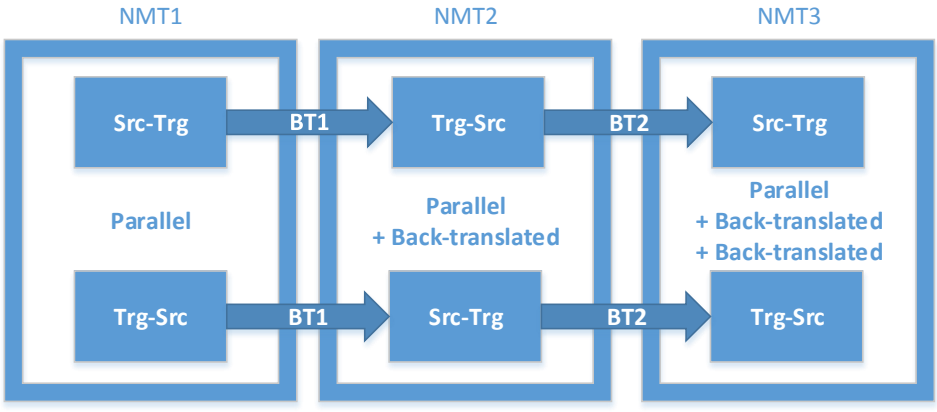


Figure 21. Multi-pass NMT training via double back-translation.

First, we trained baseline models using only filtered parallel datasets. Then, we back-translated the first batches of monolingual news data and trained intermediate NMT systems. Finally, we used the intermediate NMT systems to back-translate the second batches of monolingual news data and trained final NMT systems. The final step was performed only for En ↔ Et systems.

Automatic evaluation results using the SacreBLEU evaluation tool (Post, 2018) are given in Table 23. The results show that the multi-pass hybrid approach turned out to be the most competitive, reaching 3rd place according to automatic evaluation. Table 24 shows the manual evaluation results of the two final submissions to the shared task. The manual evaluation results show that there was no statistically significant difference between the first three En → Et systems and first seven Et → En systems, meaning that both *tilde-c-nmt-2bt* systems were tied for 1st place.

Table 23. Automatic evaluation results of the submitted systems (named *tilde-c-nmt-2bt* and *tilde-c-nmt-1bt* in the official submission) at the WMT18 shared news translation task, only considering constrained systems.

System	BLEU	
	Score	Rank
Estonian → English	28.0	7 of 23
English → Estonian	23.6	3 of 18
Finnish → English	23.0	5 of 17
English → Finnish	16.9	5 of 18

Table 24. Automatic (BLEU) and human ranking of the submitted systems (*tilde-c-nmt-2bt*) at the WMT18 shared news translation task, only considering primary constrained systems. Human rankings are shown by clusters according to Wilcoxon signed-rank test at p-level p<=0.05

System	Rank		
	BLEU	Human	
		Cluster	Ave %
Estonian → English	7 of 23	1-7 of 9	3 of 9
English → Estonian	3 of 18	1-3 of 9	3 of 9

6. Conclusions

The research conducted in this paper analyses a variety of methods for combining multiple machine translation systems. The research is mostly dedicated to combining statistical and neural machine translation methods in theoretical and practical implementations; it also includes a theoretical overview of system combinations of rule-based and other less popular machine translation paradigms. A majority of this research is focused on translation from and into Latvian, several additional experiments are performed with other morphologically rich languages, such as Czech, Estonian, Finnish, German and Russian.

The four main results are: 1) a method for hybrid MT combination using chunking and neural language models; 2) a method for hybrid NMT combination using neural network attention alignments; 3) a method for multi-pass incremental training for NMT; 4) graphical tools for overviewing and debugging the processes. The work conducted is a substantial contribution to the field of machine translation on a national and international level: 1) the author's initial idea of employing an LM to score translations and choose the best has proven to be useful even after the paradigm shift from SMT to NMT; 2) among noteworthy contributions of this work are also several state-of-the-art MT systems (Estonian ↔ Russian and Estonian ↔ English) along with details and required tools for reproducibility; 3) the tool for NMT output comprehension using attention alignments not only clearly displays the relation between the source text and the translation, but also is the first and only tool that allows the user quickly locate worse example translations to better understand shortcomings of the MT system in question.

The method for hybrid MT combination via chunking and neural language models has proven to outperform individual similar-quality systems in machine translation of texts with very long sentences. The method demonstrated good performance when working with SMT output, while for NMT output and shorter sentences the chunking method had little to poor influence. Nevertheless, even without chunking part, it is still often very useful to rescore NMT output or choose the best translation using a neural LM.

The hybrid combination method for NMT via neural network attention alignments complies with the emerging technology of neural network MT. It helps distinguish low quality resulting translations from high-quality ones without any references and use them in a hybrid combination setup. Aside from using the method for combining MT output, it has been employed in several MT quality estimation research papers (Ive et al., 2018; Yankovskaya et al., 2018).

The hybrid method of multi-pass incremental training for NMT allowed to be between the top-3 best systems in the annual news translation competition when translating into a morphologically-rich and low-resourced language – Estonian. Since the difference in human evaluation between the top-3 systems was not statistically significant (while it was statistically significant when compared to all other systems), both systems can be considered as the current state-of-the-art for Estonian ↔ English MT. The method has also proven to be competitive for systems translating into Finnish, Latvian and other complex languages and it is anticipated that it will be widely used in this year's WMT shared task for news translation.

The developed graphical tools help to inspect how translations are composed from component systems, and overview results of generated translations to locate better or worse results quickly. Aside from being useful for researchers to help them understand how systems produced specific output, these tools can also help people using public online MT systems, by outlining correlation between source and translation words. The NMT visualization and debugging tool is used to teach students in Charles University, the University of Tartu and in the University of Zurich.

References

Bahdanau, D., Cho, K., Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Banerjee, S., Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization* (Vol. 29, pp. 65-72).

Bojar, O., Chatterjee, R., Federmann, C., Haddow, B., Huck, M., Hokamp, C., Koehn, P., Logacheva, V., Monz, C., Negri, M., Post, M., Scarton, C., Specia, L., Turchi, M. (2016). Findings of the 2016 conference on machine translation. *In Proceedings of the First Conference on Machine Translation (pp. 131–198)*. Association for Computational Linguistics Berlin, Germany.

Bojar, O., Helcl, J., Kocmi, T., Libovický, J., Musil, T. (2017a). Results of the WMT17 Neural MT Training Task. In *Proceedings of the Second Conference on Machine Translation (WMT17)*, Copenhagen, Denmark.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., Monz, C., Negri, M., Post, M., Rubino, R., Specia, L., Turchi, M (2017b). Findings of the 2017 Conference on Machine Translation (WMT17). In *Proceedings of the Second Conference on Machine Translation (WMT17)*, Copenhagen, Denmark.

Cho, K., van Merriënboer, B., Bahdanau, D., Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. *Syntax, Semantics and Structure in Statistical Translation*, 103.

Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y. N. (2017). Convolutional Sequence to Sequence Learning. *Proceedings of the 34th International Conference on Machine Learning*, in PMLR 70:1243-1252

Gers, F. A., Schmidhuber, J., Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM. In *Neural Computation 12*, no. 10: 2451-2471.

Heafield, K. (2011). KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pp. 187-197. Association for Computational Linguistics.

Helcl, J, Libovický, J. (2017). Neural Monkey: An Open-source Tool for Sequence Learning. In *The Prague Bulletin of Mathematical Linguistics*, (107):5–17, 2017. ISSN 0032-6585. doi: 10.1515/pralin-2017-0001.

Hieber, F., Domhan, T., Denkowski, M., Vilar, D., Sokolov, A., Clifton, A., Post, M. (2017). Sockeye: A Toolkit for Neural Machine Translation. *Arxiv.Org*.

Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. In *Neural computation 9*, no. 8: 1735-1780.

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

Jurafsky, D., Martin, J. H. (2014). *Speech and language processing* (Vol. 3). London: Pearson.

Junczys-Dowmunt, M., Dwojak, T., Hoang, H. 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, 2016.

Klein, G., Kim, Y., Deng, Y., Crego, J., Senellart, J., Rush, A. M. (2017). OpenNMT: Open-source Toolkit for Neural Machine Translation. *Proceedings of ACL 2017, System Demonstrations*, 67–72. https://doi.org/10.18653/v1/P17-4012

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C. (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* (pp. 177-180). Association for Computational Linguistics.

Krause, B., Lu, L., Murray, I., Renals, S. (2017). Multiplicative LSTM for sequence modelling. In *5th International Conference on Learning Representations* (p. 9). Toulon, France. Retrieved from https://openreview.net/forum?id=SJCS5rXFl

Lui, M., Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations* (pp. 25–30). Association for Computational Linguistics. Retrieved from https://dl.acm.org/citation.cfm?id=2390475

Marie, B., Wang, R., Fujita, A., Utiyama, M., Sumita, E. (2018). NICT's Neural and Statistical Machine Translation Systems for the WMT18 News Translation Task. In the proceedings of *The 3rd Conference on Machine Translation*.

Och, F. J., Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational linguistics*, *29*(1), 19-51.

Paikens, P., Rituma, L., Pretkalnina, L. (2013). Morphological analysis with limited resources: Latvian example. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013)*; May 22-24; 2013; Oslo University; Norway. NEALT Proceedings Series 16, number 085, pages 267–277. Linkoping University Electronic Press.

Papineni, K., Roukos, S., Ward, T., Zhu, W.-J. (2001). BLEU. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02* (p. 311). Morristown, NJ, USA: Association for Computational Linguistics.

Pinnis, M. (2013). Context independent term mapper for European languages. In *RANLP*, pages 562–570.

Pinnis, M., Krišlauks, R., Deksne, D., Miks, T. (2017). Neural machine translation for morphologically rich languages with improved sub-word units and synthetic data. In *International Conference on Text, Speech, and Dialogue*. Springer

Pinnis, M., Rikters, M., Krišlauks, R. (2018, October). Tilde's Machine Translation Systems for WMT2018. In the proceedings of *The 3rd Conference on Machine Translation*.

Post, M. (2018). A Call for Clarity in Reporting BLEU Scores. Retrieved from http://arxiv.org/abs/1804.08771

Randolph, J. J. (2005). Free-Marginal Multirater Kappa (multirater K [free]): An Alternative to Fleiss' Fixed-Marginal Multirater Kappa. *Presented at the Joensuu Learning and Instruction Symposium*, vol. 2005.

Ramisch, C. (2012). A generic framework for multiword expressions treatment: from acquisition to applications. In *Proceedings of ACL 2012 Student Research Workshop*, pages 61–66. Association for Computational Linguistics.

Rikters, M. (2018b, September) Impact of Corpora Quality on Neural Machine Translation. In *The 8th Conference on Human Language Technologies - the Baltic Perspective (Baltic HLT 2016)*

Rikters, M., Pinnis, M., Rozis, R., Krišlauks, R. (2018b, September) Advancing Estonian Machine Translation. In *The 8th Conference on Human Language Technologies - the Baltic Perspective (Baltic HLT 2016)*

Rikters, M. (2018a, July). Debugging Neural Machine Translations. In *The 13th International Baltic Conference on Databases and Information Systems*

Rikters, M., Pinnis, M., Krišlauks, R. (2018a, May). Training and Adapting Multilingual NMT for Less-resourced and Morphologically Rich Languages. In Proceedings of *The 11th International Conference on Language Resources and Evaluation (LREC 2018)*. Paris, France: European Language Resources Association (ELRA).

Rikters, M., Fishel, M. (2017, September). Confidence Through Attention. In the proceedings of *The 16th Machine Translation Summit*.

Rikters, M., Bojar, O. (2017b, September). Paying Attention to Multi-word Expressions in Neural Machine Translation. In the proceedings of *The 16th Machine Translation Summit*.

Rikters, M., Amrhein, C., Del, M., Fishel, M. (2017, September). C-3MA: Tartu-Riga-Zurich Translation Systems for WMT17. In the proceedings of *The 2nd Conference on Machine Translation*.

Rikters, M., Fishel, M., Bojar, O. (2017a, August). Visualizing Neural Machine Translation Attention and Confidence. In *The Prague Bulletin for Mathematical Linguistics* issue 109.

Rikters, M. (2016d, December). Neural Network Language Models for Candidate Scoring in Hybrid Multi-System Machine Translation. In *CoLing 2016, 6th Workshop on Hybrid Approaches to Translation (HyTra 6)*.

Rikters, M. (2016c, October). Searching for the Best Translation Combination Across All Possible Variants. In *The 7th Conference on Human Language Technologies - the Baltic Perspective (Baltic HLT 2016)* (pp. 92-96).

Rikters, M. (2016b, September). Interactive multi-system machine translation with neural language models. In *Frontiers in Artificial Intelligence and Applications*.

Rikters, M. (2016a, July). K-Translate-Interactive Multi-System Machine Translation. In *The 12th International Baltic Conference on Databases and Information Systems* (pp. 304-318). Springer International Publishing.

Rikters, M., Skadiņa, I. (2016b, May). Syntax-based multi-system machine translation. In N. C. C. Chair) et al. (Eds.), In Proceedings of *The 10th International Conference on Language Resources and Evaluation (LREC 2016)*. Paris, France: European Language Resources Association (ELRA).

Rikters, M., Skadiņa, I. (2016a, April) Combining machine translated sentence chunks from multiple MT systems. In *The 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2016)*.

Rikters, M. (2015, July). Multi-system machine translation using online APIs for English-Latvian. In *ACL-IJCNLP 2015, 4th Workshop on Hybrid Approaches to Translation (HyTra 4)*.

Sennrich, R., Haddow, B., Birch, A. (2016a). Edinburgh Neural Machine Translation Systems for WMT 16. *Proceedings of the First Conference on Machine Translation (WMT-16)*, 2, 371–376. https://doi.org/10.18653/v1/W16-2323

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Barone, A.V.M, Mokry, J. (2017). Nematus: a Toolkit for Neural Machine Translation. *EACL 2017*, page 65, 2017.

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

Skadiņa, I., Aker, A., Giouli, V., Tufis, D., Gaizauskas, R., Mieriņa, M., Mastropavlos, N. (2010). A Collection of Comparable Corpora for Under-resourced Languages. In *Proceedings of the Fourth International Conference Baltic HLT 2010* (pp. 161-168).

Skadiņa, I., Levāne-Petrova, K., Rābante, G. (2012). Linguistically Motivated Evaluation of English-Latvian Statistical Machine Translation. In *Human Language Technologies – The Baltic Perspective - Proceedings of the Fifth International Conference Baltic HLT 2012*, IOS Press, Frontiers in Artificial Intelligence and Applications, Vol. 247, pp. 221-229.

Skadiņš, R., Goba, K., Šics, V.: Improving SMT for Baltic Languages with Factored Models. *Proceedings of the Fourth International Conference (Baltic HLT 2010)*, Frontiers in Artificial Intelligence and Applications, Vol. 2192., 125-132. (2010)

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas* (Vol. 200, No. 6).

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., Tufis, D., Varga, D. (2006). The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of LREC 2006*.

Steinberger, R., Eisele, A., Klocek, S., Pilos, S., Schlüter, P. (2012). Dgt-tm: A freely available translation memory in 22 languages. In *Proceedings of LREC 2012*.

Sutskever, I., Vinyals, O., Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems* (pp. 3104-3112).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Polosukhin, I. (2017). Attention is All you Need. In I. G. Garnett, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 5998–6008). Curran Associates, Inc. https://doi.org/10.1017/S0140525X16001837

Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J. (2016). Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *CoRR*, abs/1609.08144. URL http://arxiv.org/abs/1609.08144.

Received June 25, 2019, accepted July 10, 2019