Deeper Machine Translation and Evaluation for German

Eleftherios Avramidis, Vivien Macketanz, Aljoscha Burchardt,
Jindrich Helcl and Hans Uszkoreit
German Research Center for Artificial Intelligence (DFKI)
Language Technology Lab
firstname.lastname@dfki.de

Abstract

This paper describes a hybrid Machine Translation (MT) system built for translating from English to German in the domain of technical documentation. The system is based on three different MT engines (phrase-based SMT, RBMT, neural) that are joined by a selection mechanism that uses deep linguistic features within a machine learning process. It also presents a detailed source-driven manual error analysis we have performed using a dedicated "test suite" that contains selected examples of relevant phenomena. While automatic scores show huge differences between the engines, the overall average number or errors they (do not) make is very similar for all systems. However, the detailed error breakdown shows that the systems behave very differently concerning the various phenomena.

1 Introduction

This paper describes a hybrid Machine Translation (MT) system built for translating from English to German in the domain of technical documentation. The system builds upon the general architecture described in Avramidis et al. (2016), but in the current version several components have been improved or replaced. As detailed in the previous paper, the design of the system was driven by the assumptions that a) none of today's common MT approaches, phrase-based statistical (PB-SMT) or rule-based (RBMT), is on its own capable of providing enough good translations to be useful in an outbound translation scenario without human intervention, and b) "deep" linguistic knowledge should help to improve translation quality. Instead of building a completely new system, our goal is to adjust and combine existing systems in a smart way using linguistic knowledge.

The system has been developed within the QTLeap project[1]. The goal of the project is to explore different combinations of shallow and deep processing for improving MT quality w.r.t a real use-case scenario (translating user queries and expert answers in a chat-based PC helpdesk scenario). The system presented in this paper is the final one in a series of system prototypes developed in the project. The most visible change compared to our earlier system described in (Burchardt et al., 2016a) is that we added a neural MT system for the obvious reason that this method has shown state-of-the-art performance, e.g., in the WMT-2016 translation challenge (Bojar et al., 2016). We wanted to see how this new type of SMT engine can improve our hybrid system.

In line with our general strategy to include language experts in the MT development cycle described in Burchardt et al. (2016b), we have performed a detailed source-driven error analysis using a dedicated "test suite" that contains selected examples of relevant phenomena. Especially when using MT approaches other than SMT, this makes sure that researchers striving for insights and ideas for improvement are not discouraged by using (only) automatic scores like BLEU that are by design unable to detect changes at the needed level of detail.

This paper is organised as follows: section 2 describes the component of our architecture and section 3 our evaluation efforts. Section 4 sums up and concludes the paper.

[1] QTLeap project: http://qtleap.eu/

Proceedings of the 2nd Deep Machine Translation Workshop (DMTW 2016), pages 29–38,
Lisbon, Portugal, 21 October 2016.

2 System components

Our overall hybrid architecture includes:

- A (statistical) Moses baseline system,
- the commercial transfer-based system Lucy,
- a neural MT system, and
- an informed selection mechanism ("ranker").

The architecture is illustrated in figure 1 and the different components are described below.

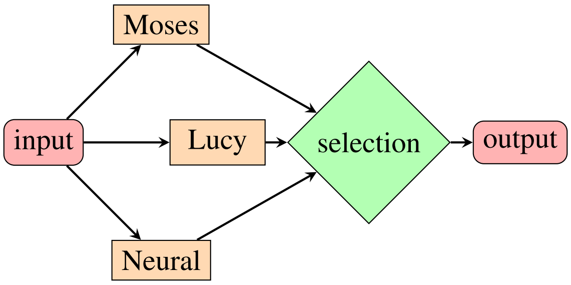

Figure 1: Architecture of the selection mechanism

2.1 Phrase-based SMT baseline

The baseline system consists of a basic phrase-based SMT model, trained with the state-of-the-art settings on both the generic and technical data. The translation table was trained on a concatenation of generic and technical data, filtering out the sentences longer than 80 words. The first batch of the QTLeap corpus[2] was used as a tuning set for MERT (Och, 2003), whereas the second batch was reserved for testing.

One language model (monolingual) of order 5 was trained on the target side from both the technical (IT-domain) and Europarl corpora, plus one language model was trained on the target-language news corpus from the years 2007 to 2013 (Callison-Burch et al., 2007). All language models were interpolated on the tuning set (Schwenk and Koehn, 2008). The size of the training data is shown in Table 1.

The text has been tokenized and truecased (Koehn et al., 2008) prior to the training and the decoding, and de-tokenized and de-truecased afterwards. A few regular expressions were added to the tokenizer, so that URLs are not tokenized before being translated. Normalization of punctuation was also included, mainly in order to fix several issues with variable typography on quotes.

The phrase-based SMT system was trained with Moses (Koehn, 2010) using EMS (Koehn, 2010), whereas the language models were trained with SRILM (Stolcke, 2002) and queried with KenLM (Heafield, 2011).

All statistical systems presented below are extensions of this system, also based on the same data and settings, unless stated otherwise.

2.2 Rule-based component

The rule-based system Lucy (Alonso and Thurmair, 2003) is also part of our experiment, due to its state-of-the-art performance in the previous years. Additionally, manual inspection on the development set has shown that it provides better handling of complex grammatical phenomena particularly when translating into German, due to the fact that it operates based on transfer rules from the source to the target syntax tree.

Additional work on RBMT focused on issues revealed through manual inspection of its performance on the QTLeap corpus (see also section 3):

[2] `http://metashare.metanet4u.eu/go2/qtleapcorpus`

corpus	entries	words
Chromium browser	6.3K	55.1K
Drupal	4.7K	57.4K
Libreoffice help	46.8K	1.1M
Libreoffice UI	35.6K	143.7K
Ubuntu Saucy	182.9K	1.6M
Europarl (mono)	2.2M	54.0M
News (mono)	89M	1.7B
Commoncrawl (parallel)	2.4M	53.6M
Europarl (parallel)	1.9M	50.1M
MultiUN (parallel)	167.6K	5.8M
News Crawl (parallel)	201.3K	5.1M

Table 1: Size of corpora used for SMT.

	BLEU	METEOR
baseline	24.90	44.38
quotes	24.00	44.29
sepMenus	25.39	45.01
sepMenus + normPunct	25.41	45.06
SMTmenus	24.06	42.83
unk	24.50	44.05
unk + sepMenus	23.68	43.30
unk + SMTmenus	25.41	44.95

Table 2: Improvements on the RBMT system measured on part of the QTLeap corpus.

- **Separate menu items**: The rule-based system was observed to be incapable of handling menu items properly, mostly when they were separated by the ">" symbol, as they often ended up as compounds. We identified the menu items by searching for consequent title-cased chunks before and after each separator. These items were translated separately from the rest of the sentence, to avoid them being bundled as compounds. The rule-based system was then forced to treat the pre-translated menu items as chunks that should not be translated.
- **Menu items by SMT**: Additionally, we used the method above to check whether menu items could be translated with the baseline phrase-based SMT system instead of Lucy.
- **Unknown words by SMT**: Since Lucy is flagging unknown words, we translated these individually with the baseline phrase-based SMT system.

Finally, we experimented with normalization of the punctuation (which was previously included in the pre-processing steps of SMT but not in RBMT), addition of quotes on the menu items and some additional automatic source pre-processing in order to remove redundant phrases such as "where it says".

We ran exhaustive search with all possible combinations of the modification above and the most indicative automatic scores are shown in table 2. Although automatic scores have in the past shown low performance when evaluating RBMT systems, our proposed modifications have a lexical impact that can be adequately measured with n-gram based metrics. Our investigation and discussion is performed on Batch 2 of the QTLeap corpus[3]. The best combination of the suggested modifications achieves an overall improvement of 0.51 points BLEU and 0.68 points METEOR over the baseline. In particular:

[3] In this paper, we will refer to Batch 1 and Batch 2 of the QTLeap corpus. This refers to the first 1000 and second 1000 sentences of the corpus.

- Adding quotes around menu items resulted in a significant drop of the automatic scores, so it was not used; this needs to be further evaluated, as references do not use quotes for menu items either. Nevertheless, quotes were not always useful due to an occasional erroneous identification of menu item boundaries.
- Separate translation of the menu items (sepMenus) gives a positive result of about 0.46 BLEU and 0.63 METEOR.
- Normalizing punctuation (normPunct) has a slightly positive effect when the menu items are translated separately by Lucy.
- Passing only RBMT's unknown words (unk) to SMT results in a loss of 0.4 BLEU.
- Translating the RBMT's menus with SMT (SMTmenus) also deteriorates the scores and
- translating both menu items and unknown words with SMT (unk+SMTmenus) has a positive effect against the baseline and it seems to be comparable with the best system without SMT (sepMenus+normPunct).

2.3 Neural MT system

Our Neural MT algorithms follow the description of Bahdanau et al. (2014). The input sequence is processed using a bidirectional RNN encoder with gated recurrent units (GRU) (Cho et al., 2014) into a sequence of hidden states. The final backward state of the encoder is then projected and used as the initial state of the decoder. Again, our decoder is composed of an RNN with GRU units. In each step, the decoder takes its hidden state and the attention vector (a weighted sum of the hidden states of the encoder, computed separately in each decoding step), and produces the next output word.

In addition to the attention model, we use byte pair encoding (Sennrich et al., 2015) in the preprocessing step. This ensures that there are no out-of-vocabulary words in the corpus and, at the same time, enables for open-vocabulary decoding.

We trained our model on the same data as the PBMT baseline system. We used Batch 1 for validation during the training. In the experiments, the sentence length was limited to 50 tokens. The size of the hidden state of the encoder was 300 units, and the size of the hidden state of the decoder was 256 units. Both source and target word embedding vectors had 300 dimensions. For training, a batch size of 64 sentences was used. We used dropout and L2 for regularization.

Our model was implemented using Neural Monkey,[4] a sequence to sequence learning toolkit built on top of the Tensorflow framework (Abadi et al., 2016). This toolkit was used before by Libovický et al. (2016) in the submission of WMT-2016's multimodal translation and automatic post-editing tasks.

2.4 Selection mechanism

The three systems above are combined with a selection on the sentence level. For every source sentence, the output of every available system is analyzed with several automatic NLP techniques to produce numerical values which indicate some aspects of quality. Out of the numerical values, we form one feature vector which represents the qualitative characteristics of every produced translation output. Consequently, we employ an empirical mechanism which aims to *rank* and *select* given these feature vectors.

The core of the selection mechanism is a ranker which reproduces ranking by aggregating pairwise decisions by a binary classifier (Avramidis, 2013). Such a classifier is trained on binary comparisons in order to select the best one out of two different MT outputs given one source sentence at a time. As training material, we use the test-sets of WMT evaluation task (2008-2014). The rank labels for the training are given by human annotators, as part of the WMT evaluation campaign. The binary comparisons are aggregated per system and the winner is the system which wins the most pairwise comparisons. In order to eliminate cases where two systems win an equal number of pairwise comparison, we weigh each pairwise comparison with its confidence score (soft pairwise recomposition) (Avramidis, 2013).

[4] `http://github.com/ufal/neuralmonkey`

We exhaustively tested the available feature vectors with many machine learning methods including Naïve Bayes, k-nearest Neighbors, Logistic Regression, Linear Discriminant Analysis, Extremely Randomized Trees, Random Tree Forests, Bagging Classifiers, AdaBoost and Gradient Boosting. The models produced were scored in terms of correlation with the original human ranks with Kendall's tau. The scoring was performed on a cross-validation with 10 folds over the entire amount of WMT data. The best correlation was given with Gradient Boosting over an ensemble of 100 single Decision Tree classifiers.

The feature vector consists of 56 distinct features including:

- **Parse probabilities**: the number of feasible k-best parse trees, the highest and the lowest probability in the k-best parse tree list, the mean and the standard deviation of the parse probabilities in the k-best parse tree list,
- **Parse nodes**: the distance of main and subordinate VPs from the end of the target sentence, the count and the average position of nouns in the sentence, the count and the standard deviation of the positions of NPs, PPs and VPs in the sentence, the average and maximum height of VPs in the parse tree, count of target NPs aligned with source NPs via IBM model 1,
- **Punctuation and case**: count, average position and standard deviation of commas, count of dots, uppercase sentence start,
- **Contrastive scores**: BLEU and METEOR using the other two systems as references,
- **Language modeling**: 5-gram language model probability,
- **IBM model 1**: the IBM model 1 scores on both directions, and their ratios, thresholded by either 0.01 and 0.2,
- **Baseline features**: the baseline features of WMT12.

All features can be reproduced using the tool Qualitative (Avramidis, 2016).

3 Evaluation

As mentioned earlier, the hybrid system is developed and tested on a technical domain. All results following are done on the second batch of the QTLeap corpus.

3.1 Automatic evaluation results

Table 3 shows BLEU scores for the systems that we finally chose to feed into our selection mechanism. As reference, we also show scores for the baseline phrase-based SMT system without any of improvements described above. For evaluation, we have used MTComparEval (Klejch et al., 2015). While BLEU scores suggest that both statistical systems are clearly outperforming the RBMT system and also the selection mechanism, inspection of examples did not show such a clear picture.

	BLEU
PB-SMT	37.43
RBMT-baseline	26.96
RBMT-improved	27.53
Neural MT	39.02
Selection mechanism	31.03

Table 3: Automatic scores of the independent components and the selection mechanism

In order to get more insights into the strengths and weaknesses of the systems, we performed a more systematic and detailed manual error analysis on the results.

3.2 Manual evaluation

In context of the QT21 project[5], we have constructed an expansive test suite containing a wide range of various linguistic phenomena that provides a basis for manual analyses in different contexts. Depending on the focus of a manual inspection, different subsections of the test suite can be used to test and evaluate systems. Inspired by the performance of the systems reported here on the test suite, we have constructed a domain-specific test suite based on examples from the QTLeap corpus that represent interesting linguistic phenomena.

The "linguistic phenomena" are understood in a pragmatic sense and cover various aspects that influence the translation quality. Therefore, our phenomena include morpho-syntactic and semantic categories as well as formatting issues, issues of style, etc.

3.3 Manual evaluation methodology

Starting from our evaluation in our contribution to the WMT2016 IT task (Avramidis et al., 2016), we have by now developed an efficient manual evaluation process, performed by a professional German linguist. This procedure consists of the following steps:

1. The linguist has a close look at the output of the different MT systems and identifies systematically occurring translation errors that are related to linguistic phenomena.

2. For each of these linguistic phenomena that seem to be prone to translation errors, 100 segments containing the phenomenon in the source language are extracted, as inspecting the complete test set would be too time-consuming.

3. For each phenomenon, the total occurrences in the source language are counted.

4. Consequently, the total occurrences in the outputs of the different MT systems are counted.

5. The accuracy of the MT outputs for the phenomena is measured by dividing the overall number of correctly translated instances by the overall number of instances in the source segments.

The phenomena that we found to be prone to translation errors in this context were **imperatives**, **compounds**, **menu item separators** (separated by ">"), **quotation marks**, **verbs**, **phrasal verbs** and **terminology**.

As there may always be several correct translations, an occurrence of a phenomenon is not only counted as correctly translated when it matches the reference translation but also when it is for example realized in a different structure that correctly translates the meaning. The following examples demonstrate the manual evaluation technique:

(1)	source:	Yes, type, for example: 50 miles in km.	*1 inst.*
	PB-SMT:	Ja, Typ, zum Beispiel, 50 Meilen in km.	*0 inst.*
	neural:	Ja, Typ, beispielsweise: 50 Meilen in km.	*0 inst.*
	RBMT-imp.:	Tippen Sie zum Beispiel, ja: 50 Meilen in km.	*1 inst.*
	reference:	Ja, geben Sie, zum Beispiel: 50 Meilen in km ein.	

In example 1, the source segment contains one imperative: "type". A correct German translation needs to have the right verb from + the personal pronoun "Sie" in this context. In most of the cases, the imperative "type" is mistranslated as the German noun "Typ" instead of the verb "tippen" or "eingeben", e.g., in the PB-SMT and neural output. The improved RBMT system on the other hand correctly translates the imperative. Note that the reference translation contains the phrasal verb "eingeben" and due to the imperative construction the suffix "ein" moves to the end of the sentence.

[5] www.qt21.eu

(2) source:	[...] Adjustments ≥ Notification Center ≥ Mail.	*2 inst.*
PB-SMT:	[...] Adjustments>-Benachrichtigungszentrale ≥ E-Mail.	*1 inst.*
RBMT:	[...] Anpassungs->-Benachrichtigungs-Zentrums->-Post [...].	*0 inst.*
RBMT-imp.:	[...] Anpassungen ≥ Benachrichtiungs-Zentrum ≥ Post [...].	*2 inst.*
reference:	[...] Anpassungen ≥ Benachrichtigungszentrum ≥ Post [...].	

Example 2 depicts the analysis of the menu item separators. The source contains two instances. The PB-SMT output treats the words before and after the first separator as a compound, adding a hyphen after the separator. Therefore, only the second separator counts as correct. The RBMT treats the separators similarly, adding hyphens before and behind the separators, resulting in no correct instances. The improved RBMT version treats all separators correctly.

3.4 Manual evaluation results

The manual evaluation for the paper at hand includes the five systems described above: PB-SMT, RBMT, RBMT-improved, the neural system and the selection mechanism. For the aforementioned seven linguistic phenomena, 657 source segments were extracted[6]. In those 657 source segments, 2105 instances of the different phenomena were found overall, as it was often the case that more than one instance occurred per segment. The results appear in table 4.

	#	PB-SMT	RBMT	RBMT improved	neural	sel. mech.
imperatives	247	68%	79%	**79%**	74%	*73%
compounds	219	55%	87%	**85%**	51%	70%
">" separators	148	**99%**	39%	83%	93%	80%
quotation marks	431	**97%**	94%	75%	95%	80%
verbs	505	85%	93%	**93%**	90%	*90%
phrasal verbs	90	22%	68%	**77%**	38%	53%
terminology	465	**64%**	50%	53%	55%	54%
sum	2105					
average		76%	77%	77%	75%	74%

Table 4: Translation accuracy on manually evaluated sentences focusing on particular phenomena. Testsets consist of hand-picked source sentences that include the respective phenomenon. Simple RBMT is separated as it does not participate in the selection mechanism. The percentage of the best system in each category is bold-faced, whereas (*) indicates that there is no significant difference ($\alpha = 0.05$) between the selection mechanism and the best system.

As it can be seen in the table, the overall average performance of the components is very similar with no statistically significant difference. The phrase-based SMT and the RBMT system have the highest overall average scores but interestingly their performances on the different linguistic phenomena are quite complimentary:

While the baseline **PB-SMT** system operates best of all systems on the menu item separators (">"), the quotation marks and terminology, the baseline **RBMT** system performs best on the remaining linguistic categories, namely the imperatives, compounds, verbs and phrasal verbs, as well as the quotation marks. The PB-SMT system is furthermore doing well on imperatives and verbs but it has the lowest score of all systems regarding the phrasal verbs. The RBMT system on the other hand also reaches a high score for the quotation marks but has the lowest scores for the menu item separators.

The improved version of the RBMT system, namely the **RBMT-improved**, has the same performance in the overall average compared to its base system. Likewise, it ranks among the best-performing systems

[6] Despite the goal of collecting 100 segments per category, it was only possible to find 57 segments with phrasal verbs.

in terms of imperatives, compounds, verbs and phrasal verbs. Furthermore, it significantly improved on the category it was developed for, i.e. the menu item separator “>”. At the same time it has visibly lower scores for the quotation marks (as a side effect of the improved treatment of menu items, the treatment of quotation marks is much worse than for the RBMT baseline system).

The **neural** system reaches a slightly lower score than the other systems. It ranks among the best systems regarding the imperatives, quotation marks and verbs. Furthermore it also shows high scores for the menu item separators. Its score for the compounds on the other hand is the lowest of all systems, close to that of the phrase-based SMT.

The **selection mechanism** obtains the lowest average value of all systems but this score is only three percentage points less than the highest average value. The selection mechanism is one of the best performing systems on imperatives and verbs. For the other phenomena it mostly reaches a score that is lower than the scores of its component systems.

4 Discussion and further work

In this paper, we have presented a hybrid machine translation architecture for German that is based on three different MT engines (phrase-based SMT, RBMT, neural) that are joined by a selection mechanism that uses deep linguistic features among other things.

In terms of evaluation, we have also taken a “deep” approach by integrating a linguist who evaluated the systems’ errors on a kind of test suite made of relevant examples representing seven selected “error categories” such as imperatives or terminology. While overall automatic BLEU results indicate that the systems differ in performance (statistical systems outperform the selection mechanism who outperforms the rule-based system), the detailed error evaluation on segment basis has shown a different picture. Looking at the bottom line, i.e., the sum of selected error categories, it seems that all systems perform alike: they all get about 75% of the examples (“error triggers”) right. But if we look at the detailed distribution, it turns out that the systems perform significantly different on the different categories.

While the selection mechanism was the best performing system in terms of errors in previous work (Avramidis et al., 2016), it was not able to reduce some of the errors in the given experiments. This is not surprising as the error categories shown here are not explicitly handled by the selection mechanism. Additionally, whereas the performance is measured on a domain-specific test-set, the selection mechanism was trained on generic news corpora, for lack of in-domain annotations. As future work, one option would be to explicitly handle the errors, e.g., by including features into the selection mechanism that trigger certain action if, say, an imperative is detected. Likewise, this error-driven type of evaluation can also be used to improve the data-driven engines, e.g., by creating special corpora for selected phenomena.

Whenever dealing with language, it is clear that phenomena do not act independently, thus it is not surprising that the improvement of one phenomenon may have side effects on the other phenomena (as it was the case with Lucy-improved). For future work, we plan to partly automate our test suites in order to be able to track the effects of improving the systems. For the time being, this can also be realized by manual inspection with certain intermediate steps.

One notable insight of the experiments presented here and also from further inspection of examples is that the neural system does not seem to be able to take advantage of the (technical) domain data in the training data as compared to phrase-based SMT. This might explain why the neural system does not massively outperform phrase-based SMT here as it has been seen in other contexts cited above.

Acknowledgments

This work has received support from the ECs FP7 (FP7/2007-2013) under grant agreement number 610516 (QTLeap) and from the ECs Horizon 2020 research and innovation programme under grant agreement number 645452 (QT21).

References

Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, and Others. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Juan A Alonso and Gregor Thurmair. 2003. The Comprendium Translator system. In *Proceedings of the Ninth Machine Translation Summit*. International Association for Machine Translation (IAMT).

Eleftherios Avramidis, Burchardt, Aljoscha, Vivien Macketanz, and Ankit Srivastava. 2016. DFKI's system for WMT16 IT-domain task, including analysis of systematic errors. In *Proceedings of the First Conference on Machine Translation*, pages 415–422, Berlin, Germany, aug. Association for Computational Linguistics.

Eleftherios Avramidis. 2013. Sentence-level ranking with quality estimation. *Machine Translation*, 27(Special issue on Quality Estimation):239–256.

Eleftherios Avramidis. 2016. Qualitative: Python Tool for MT Quality Estimation Supporting Server Mode and Hybrid MT. *The Prague Bulletin of Mathematical Linguistics (PBML)*, 106:147–158.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural {Machine} {Translation} by {Jointly} {Learning} to {Align} and {Translate}. *arXiv:1409.0473 [cs, stat]*, sep.

Ondej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 Conference on Machine Translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany, aug. Association for Computational Linguistics.

Aljoscha Burchardt, Kim Harris, Georg Rehm, and Hans Uszkoreit. 2016a. Towards a systematic and human-informed paradigm for high-quality machine translation. In Georg Rehm, Aljoscha Burchardt, Ondrej Bojar, Christian Dugast, Marcello Federico, Josef van Genabith, Barry Haddow, Jan Haji, Kim Harris, Philipp Khn, Matteo Negri, Martin Popel, Lucia Specia, Marco Turchi, and Hans Uszkoreit, editors, *Proceedings of the LREC 2016 Workshop Translation Evaluation: From Fragmented Tools and Data Sets to an Integrated Ecosystem*, pages 35–42, Portoro, Slovenia, May.

Aljoscha Burchardt, Kim Harris, Georg Rehm, and Hans Uszkoreit. 2016b. Towards a Systematic and Human-Informed Paradigm for High-Quality Machine Translation. In Georg Rehm, Aljoscha Burchardt, Ondrej Bojar, Christian Dugast, Marcello Federico, Josef van Genabith, Barry Haddow, Jan Hajic, Kim Harris, Philipp Koehn, Matteo Negri, Martin Popel, Lucia Specia, Marco Turchi, and Hans Uszkoreit, editors, *Proceedings of the LREC 2016 Workshop "Translation Evaluation: From Fragmented Tools and Data Sets to an Integrated Ecosystem". Translation Evaluation: From Fragmented Tools and Data Sets to an Integrated Ecosystem (LREC-2016), located at International Co*. o.A.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. 2007. (Meta-) evaluation of machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation (StatMT'07)*, pages 136–158, Prague, Czech Republic, jun. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar, oct. Association for Computational Linguistics.

Kenneth Heafield. 2011. KenLM : Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, number 2009, pages 187–197, Edinburgh, Scotland, jul. Association for Computational Linguistics.

Ondrej Klejch, Eleftherios Avramidis, Aljoscha Burchardt, and Martin Popel. 2015. MT-ComparEval: Graphical evaluation interface for Machine Translation development. *The Prague Bulletin of Mathematical Linguistics*, 104(1):63–74.

Philipp Koehn, Abhishek Arun, and Hieu Hoang. 2008. Towards better Machine Translation Quality for the German-English Language Pairs. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 139–142, Columbus, Ohio, jun. Association for Computational Linguistics.

Philipp Koehn. 2010. An Experimental Management System. *The Prague Bulletin of Mathematical Linguistics*, 94(-1):87–96.

Jindich Libovický, Jindich Helcl, Marek Tlustý, Pavel Pecina, and Ondej Bojar. 2016. CUNI System for WMT16 Automatic Post-Editing and Multimodal Translation Tasks. *CoRR*, abs/1606.0.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167, Morristown, NJ, USA. Association for Computational Linguistics.

Holger Schwenk and Philipp Koehn. 2008. Large and Diverse Language Models for Statistical Machine Translation. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II.*

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural Machine Translation of Rare Words with Subword Units. *CoRR*, abs/1508.0.

Andreas Stolcke. 2002. SRILM an Extensible Language Modeling Toolkit. In *System*, volume 2, pages 901–904. ISCA, sep.