Effective Online Learning Implementation for Statistical Machine Translation

Toms Miks, Mārcis Pinnis(✉), Matīss Rikters, and Rihards Krišlauks

Tilde, Vienības gatve 75A, Riga 1004, Latvia
{toms.miks,marcis.pinnis,matiss.rikters,rihards.krislauks}@tilde.lv

Abstract. Online learning has been an active research area in statistical machine translation. However, as we have identified in our research, the implementation of successful online learning capabilities in the Moses SMT system can be challenging. In this work, we show how to use open source and freely available tools and methods in order to successfully implement online learning for SMT systems that allow improving translation quality. In our experiments, we compare the baseline implementation in Moses to an improved implementation utilising a two-step tuning strategy. We show that the baseline implementation achieves unstable performance (from −6 to +6 BLEU points in online learning scenarios and over −6 BLEU points in translation scenarios, i.e., when post-edits were not returned to the SMT system). However, our devised two-step tuning strategy is able to successfully utilise online learning capabilities and is able to improve MT quality in the online learning scenario by up to +12 BLEU points.

Keywords: Phrase-based statistical machine translation
Online learning · Dynamic adaptation

1 Introduction

When working on the translation of documents or larger translation projects, it can easily become annoying if machine translation (MT) systems make the same mistakes on words, phrases, or sentences that were corrected by the translator a few segments earlier. To address this issue for MT systems, researchers have developed online learning (OL) methods that allow improving the translation quality during runtime by learning from corrected translations, which are sent back to the MT system from computer-assisted translation (CAT) tools after translators have approved a post-edited translation.

In this work, we analyse the effectiveness of online learning for two language pairs, for which MT online learning has not been previously applied – English-Estonian and English-Latvian. We build upon the implementation by Bertoldi [3], however, we show that the baseline implementation is sub-optimal and in Sect. 5 we propose a better SMT system model weight tuning strategy that allows us to develop systems of higher translation quality. Compared to

A. Lupeikiene et al. (Eds.): DB&IS 2018, CCIS 838, pp. 302–313, 2018.
https://doi.org/10.1007/978-3-319-97571-9_24

related work, we also evaluate the online learning method on large datasets consisting of 20,000 to 60,000 segments that correspond to approximately 375,000 and 475,000 running tokens respectively.

The paper is further structured as follows: Sect. 2 briefly describes relevant related work on online learning, Sect. 3 describes the data used in our experiments, Sect. 4 describes the baseline systems, Sect. 5 proposes the improved tuning strategy, Sect. 7 analyses translation memory (TM) influence in online learning scenarios, and Sect. 8 concludes the paper.

2 Related Work

Online learning has been studied for both statistical machine translation (SMT; [3,12,18]) and neural machine translation (NMT; [20,21,25]). Although NMT can be considered to be state-of-the-art for broad domain MT (as shown by its dominance in the WMT shared tasks in news translation since 2016 [6,7]), there are translation scenarios where NMT still does not out-perform SMT system translation quality. For instance, when dealing with low-resource language pairs or (more or less) controlled languages (or domains with a limited vocabulary), NMT systems have shown not to perform better than SMT systems [23]. As such scenarios (where limited datasets for narrow domains are available) are frequent for localisation service providers, this work will focus on improving the online learning methods for SMT systems.

Related work on online learning for SMT has been previously focused on the development of methods that introduce dynamic translation and language models along the static models of SMT systems and methods that build translation and language models that are designed to be dynamically adapted using post-edited translations. For instance, Bertoldi [3] introduced cache-based models for online learning in the Moses [15] SMT system. Such models have shown to allow improving translation quality in online learning scenarios if the text that needs to be translated contains repetitions (e.g., repeated words, phrases, or even sentences) [4]. There has been also considerable effort spent on identifying methods that optimise hyper-parameters SMT systems with static and dynamic translation and language models. E.g., Mathur and Cettolo [17] in their work used two optimisation methods – the Downhill Simplex Method and the Modified Hill Climbing –, however, they showed limited quality improvements (by just up to 0.5 BLEU points) in their experiments when comparing systems without online learning and with online learning.

Germann [12] describes suffix-array based translation models for the Moses toolkit. In the online learning scenario, the models are supplemented with phrase translations that are extracted from post-edited sentences. Although being a promising method, the author reported that there was insufficient evidence that the method provides better translation quality compared to a baseline system without online learning. [18] introduced a feature into a suffix-array based translation model that indicates whether phrase candidates are added from the post-edited data. The model weights are continuously tuned during online learning

in order to learn that the additional feature is important. The authors report a significant (5 BLEU point) improvement in their experiments. A similar method using the cdec [11] SMT system has been proposed by Denkowski et al. [9] where the authors successfully used a suffix-array based translation model and a cache-based language model for SMT system online adaptation. The authors report an MT error reduction of 13%.

The effectiveness of online learning in post-editing projects has been recently analysed by Bentivogli et al. [2] where the authors compared static SMT systems to SMT systems with online learning capabilities. The authors showed that the post-editing effort when using online learning could be reduced by up to 10%. However, the authors experimented on very small datasets consisting of just up to 300 sentences and the same dataset was post-edited by the translators twice - the first time with the static SMT systems and the second time with the adaptive SMT systems. Although there was a one-month time difference between the post-editing sessions, the translator performance may still be affected by remembering the translations to some extent (or at least remembering how the translation was performed).

3 Data

We evaluate the online learning method on datasets from two domains (information technologies (IT) and medicine) and for two language pairs (English-Estonian and English-Latvian). For training of the medical domain systems, we use publicly available data - the parallel corpus of the European Medicines Agency (EMEA; [24]) that primarily consists of drug (medicine) descriptions. For training of the IT domain systems, we use a collection of publicly available corpora (e.g., the Microsoft User Interface Translations parallel corpus [19]) and proprietary corpora (e.g., software documentation, user interface strings, etc.). Note that the IT domain corpora are of broader coverage in terms of vocabulary and writing styles than the medical domain corpus, which is mostly constructed from medicine descriptions. The training data statistics are summarised in Table 1. It is evident that both domains present different MT scenarios - the medical domain scenario is a low resource and narrow domain scenario, whereas the IT domain scenario is a high resource and broad domain scenario.

Table 1. Training data statistics

	IT domain		Medical domain
	English-Estonian	English-Latvian	English-Latvian
Unique parallel sentence pairs	9,059,100	4,029,063	325,332
Unique in-domain monolingual sentences	34,392,322	1,950,266	332,652
Unique broad domain monolingual sentences	-	2,369,308	-
Tuning data	1,990	1,837	2,000

For evaluation of the online learning method, we use data from two large post-editing projects - a commercial post-editing project for a private customer in the IT domain (for English-Latvian and English-Estonian), and a research post-editing project in the medical domain (for English-Latvian). The post-edited data for the experiments in the medical domain have been produced within the QT21 project[1] [22]. The IT domain data were prepared by post-editing MT translations of software documentation, user interface strings, and (IT product related) marketing texts within the MemSource[2] web-based computer-assisted translation (CAT) tool. The translations were provided by a phrase-based SMT system that was trained on a similar corpus as the training data that were used in our experiments. The medical domain data were prepared by post-editing MT translations of medicine descriptions from the EMEA home page using the Post-editing Tool (PET; [1]). The SMT system that prepared the initial translations was trained on the same training data that were used in our experiment. The training data do not include documents from the online learning evaluation set, however, there may be individual sentences that appear in the training data (we believe, that this allows to better simulate real-life translation situations where some sentences tend to be repetitive). The post-edited data statistics are given in Table 2.

Table 2. Post-edited data statistics

	IT domain		Medical domain
	English-Estonian	English-Latvian	English-Latvian
Segments	60 630	27 122	20 286
Tokens	475 295	166 350	374 914

4 Baseline Implementation

We started our experiments by training baseline SMT systems and SMT systems with baseline dynamic learning models. All MT systems were trained using the Moses SMT system on the Tilde MT[3] platform [26]. For word alignment, we used fast-align [10]. All SMT systems feature 7-gram translation models and the *wbe-msd-bidirectional-fe-allff* reordering models. The systems have either one or two language models (depending on the availability of broader domain data) that were trained using KenLM [14]. For English-Latvian, we trained 5-gram language models and for English-Estonian (due to a significantly larger monolingual corpus) - 4-gram language models. The systems were tuned using MIRA [13] on the respective tuning datasets.

[1] More information about the QT21 project can be found online at http://www.qt21.eu/.
[2] www.memsource.com.
[3] www.tilde.com/mt.

The online learning set-up is based on the implementation by Bertoldi [3]. First, the static SMT system's models are trained, after which a dynamic translation model and a dynamic language model are added to the SMT system. The system's model (both static and dynamic) log-linear weights are tuned using MIRA by iteratively translating the tuning dataset using the online learning procedure. The online learning procedure during translation is as follows:

1. The SMT system receives a translation request to translate a sentence.
2. The sentence is translated by the SMT system and the translation is sent to a CAT tool.
3. The translation is post-edited by a translator in the CAT tool.
4. The post-edited translation together with the source sentence is sent back to the SMT system to perform online learning.
5. The SMT system performs word alignment between the source sentence and the post-edited sentence using fast-align. For this, we use the fast-align model acquired during training of the SMT system.
6. The SMT system extracts parallel phrases consisting of 1-7 tokens using the Moses phrase extraction method [16] that is implemented in the Moses toolkit.
7. The extracted phrases are added to the dynamic translation and language models so that, when translating the next sentence, the system would benefit from the newly learned phrases. Phrases that are added to the dynamic models are weighted according to their age (newer phrases have a higher weight) using the hyperbola-based penalty function [3]. A maximum of 10,000 phrases is kept in the dynamic models.

In our experiments we distinguish three types of translation scenarios:

1. The baseline scenario uses a standard SMT system with no dynamic models.
2. The *OL−* scenario uses an SMT system with dynamic models, however, post-edited translations are not sent back to the SMT system for online learning. This means that the dynamic models will always stay empty. The goal of this scenario is to validate whether SMT systems with dynamic models are able to reach baseline translation quality in situations when some CAT tools are not able to or do not provide functionality that allows returning post-edited translations back to the SMT system.
3. The *OL+* scenario uses an SMT system with dynamic models and after translation of each sentence, the post-edited translation is sent back to the SMT system for online learning.

Note that having a translator ready for every experiment is expensive and time-consuming. Therefore, online learning is evaluated in a simulated online learning scenario where instead of the (dynamic) post-edits, which should be prepared by a translator when using an online learning enabled SMT system, we use (static) post-edits that were collected in the post-editing projects, where translators used a static SMT system (without online learning capabilities).

The baseline systems were evaluated using BLEU on the full post-edited datasets. Evaluation results are given in Table 3. For English-Estonian we trained

only the baseline SMT system, because we started our experiments for English-Latvian and validated only the best performing set-up for English-Estonian. The *OL−* system results show that the addition of the dynamic models negatively impacted translation quality even though the dynamic models were kept empty (for more details on why this happened, see Sect. 5). However, the translation quality does improve for the broader IT domain *OL+* system by 6 BLEU points when compared to the baseline. This means that the negative effects introduced by the dynamic models can be overcome by online learning over time. For the medical domain, the quality of the *OL+* system dropped by over 7 BLEU points. We believe that this may be caused by the high quality of the baseline system and the fact that the narrow domain data are well represented in the training dataset.

Table 3. Baseline system evaluation results

System	IT domain		Medical domain
	English-Estonian	English-Latvian	English-Latvian
Baseline	26.80 ± 0.17	26.42 ± 0.23	76.78 ± 0.17
OL−	-	19.91 ± 0.20	70.27 ± 0.20
OL+	-	32.42 ± 0.30	69.53 ± 0.22

5 Two-Step Tuning

The *OL−* systems showed a significant drop in translation quality when the post-edited sentences were not used for online adaptation (i.e., if the dynamic models were kept empty). Therefore, we looked into the tuning process and identified that when tuning all log-linear model weights together (i.e., the static model and the dynamic model weights), the tuning method did not find optimal weights for the static models. This led to the significant drop in translation quality of over 6 BLEU points for both the medical and IT domain datasets.

To address the issue, we devised a two-step tuning procedure where the static and dynamic model weights were tuned separately. The tuning method works as follows:

1. First, the static model weights are tuned using MIRA in a standard translation scenario (without online learning).
2 Then, the dynamic model weights are tuned using MERT [5] in an online learning scenario using the pre-trained static model weights. The static model weights are kept unchanged. During dataset analysis, we observed that the repetition rates [8] for the tuning and test datasets differ. We artificially increase the repetition rate in the tuning dataset to more closely match that of the test dataset, which, as our experiments showed (see Sect. 6), increases system performance. As the tuning datasets are random held-out datasets from the training data, we duplicated every n^{th} sentence in order to introduce repetitiveness in the data. For the IT domain experiments, every fourth

sentence was duplicated, and for the medical domain experiments, every sixteenth sentence was duplicated. The duplication rate differs as the medical domain data are much more narrow and they contain higher repetitiveness before duplication than the IT domain data.

The two-step tuning procedure ensures that even if the dynamic models and online learning will not be used (for instance, if a particular CAT tool that a translator uses to post-edit MT translations is not able to send the post-edited translations back to the SMT system), the SMT system will perform as good as the baseline system without any dynamic models.

The evaluation results in Table 4 show that the system quality in both scenarios (*OL−* and *OL+*) is improved by a large margin over the respective baseline systems (see Table 3). For instance, the quality of the English-Latvian IT domain *OL+* system gained 6.17 BLEU points over the respective baseline. Although in the medical domain the *OL+* system did not show an improvement in comparison to the *OL−* system, the quality decrease (−0.55 BLEU points) is much lower compared to the baseline *OL+* system's quality decrease (−7.25 BLEU points).

Table 4. Evaluation results for systems with two-step tuning

System	IT		Medical
	English-Estonian	English-Latvian	English-Latvian
Baseline	26.80 ± 0.17	26.42 ± 0.23	76.78 ± 0.17
OL−	26.80 ± 0.17	26.42 ± 0.23	76.78 ± 0.17
OL+	31.45 ± 0.20	38.59 ± 0.31	76.23 ± 0.19

6 Text Repetitiveness in the Tuning Dataset

The potential benefit of online learning depends on how much repetition is evident in the translatable content. Text repetition is also necessary for tuning data in order to successfully tune the dynamic translation and language model weights of the SMT systems. Therefore, in this section, we analyse the level of text repetitiveness required in the tuning dataset to achieve higher MT quality in online learning scenarios. We limit these experiments to the English-Latvian language pair.

As explained above, to simulate text repetitiveness in the tuning dataset, we duplicate every n^{th} (first, fourth, eight, or sixteenth) sentence pair, thereby creating four different tuning datasets. Using these datasets to tune the dynamic model weights, different weight values were identified (giving more or less strength to the dynamic models). Then, evaluation datasets were translated in the *OL+* scenario.

Evaluation results in Table 5 indicate that for the medical domain, better results are attained when every 16^{th} sentence in the tuning dataset was repeated. On the other hand, for the IT domain, the best results were attained when repeating every fourth sentence.

Table 5 provides also scores that analyse how much repetition is present in the tuning datasets and for reference also for the evaluation datasets using two text repetitiveness metrics. The first is the Repetition Rate (RR) metric [8]. The RR metric calculates text repetitiveness by analysing the number of n-grams (from 1 to 4 tokens) repeated in the text. Our observations showed that the repetition of 4-grams in the evaluation data was relatively low. Therefore, we devised a modified text repetitiveness metric – RR1 – that considers only unigrams, bigrams and trigrams. RR calculates text repetitiveness by analysing the text as a whole, thereby ignoring sentence boundaries. Since MT systems operate on a sentence-level, we restricted the RR1 metric to count n-gram repetitiveness only within sentence boundaries (and not between sentences).

The results show that for medical domain data, the highest MT quality is achieved when the repetitiveness in the tuning data is similar to the repetitiveness in the evaluation data (according to both metrics – RR and RR1). The situation slightly differs for IT domain data. It is evident that the tuning data and evaluation data RR scores differ for the configuration that achieves the best results. However, the RR1 scores of the tuning data for the best performing configuration are the most similar to the RR1 scores of the evaluation data.

The results indicate that in order to achieve the highest MT quality in online learning scenarios, the text repetitiveness according to the RR1 metric in tuning data has to be similar to the text repetitiveness in the evaluation data. The tendency is clearer when plotting the results in a chart (see Fig. 1).

Table 5. Translation quality results for English-Latvian in the *OL+* scenario for different levels of text repetitiveness in the tuning datasets.

Experiment	Medical domain			IT domain		
	RR1	RR	BLEU	RR1	RR	BLEU
Evaluation data	0.13	0.11	-	0.31	0.20	-
Tuning data – 100% repetitiveness	0.51	0.85	75.98 (75.61-76.33)	0.51	0.85	36.01 (35.42-36.63)
Tuning data – 25% repetitiveness	0.27	0.31	76.17 (75.81-76.54)	0.26	0.30	38.59 (38.01-39.21)
Tuning data – 12.5% repetitiveness	0.19	0.21	76.18 (75.83-76.54)	0.19	0.20	38.38 (37.77-38.95)
Tuning data – 6.25% repetitiveness	0.14	0.16	76.23 (75.88-76.6)	0.14	0.15	38.38 (37.79-39.02)

7 Translation Memory Influence

The Tilde MT platform provides a translation memory feature for all MT systems that send post-edited translations back to the platform. This means that during online learning scenarios, for sentences, for which full match sentences can be found, the translations are looked-up in the translation memory of the

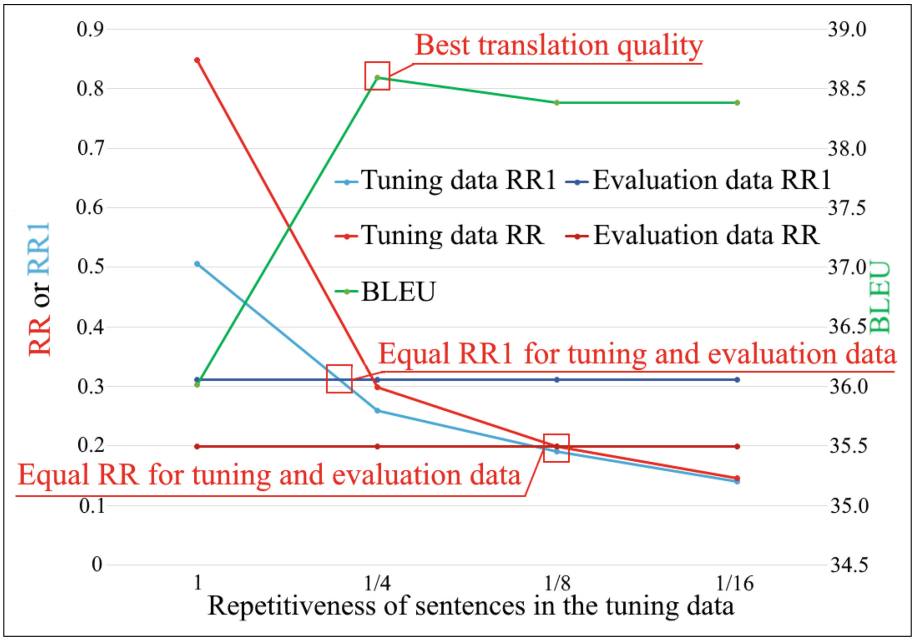

Fig. 1. Difference of translation quality for different text repetitiveness levels in the tuning data

SMT system. This means that it is important to identify, how large improvement in online learning scenarios is obtained from the translation memory alone and how large improvement can be attributed to the online learning method itself.

When analysing the post-edited data, we identified that there is a sentence level repetitiveness of 17.8% and 15% in the English-Latvian and English-Estonian IT domain post-edited datasets respectively. The repetitive segments on average consist of 2.1 and 2.5 English words in the respective datasets. All unique segments on average consist of 6.3 and 7.1 words respectively, which means that the repetitive segments are mostly short phrases (e.g., repetitive menu item, button, and label titles, etc.). The repetitiveness in the medical domain corpus was 0% due to how the data for post-editing was prepared, therefore, this analysis was not performed on the medical domain dataset.

To analyse the impact of the translation memory on the translation quality, we performed an additional experiment where post-edited sentences were used to fill the translation memory, but the online learning functionality was disabled. The results of the experiment are given in Table 6. It is evident that the translation memory accounts for 2 to 2.5 BLEU points for both language directions. However, the improvement from online learning is still substantial (9.62 and 2.58 BLEU points for English-Latvian and English-Estonian respectively over using just the translation memory). The cumulative improvement of 12.17 and 4.65 BLEU points for English-Latvian and English-Estonian respectively shows that

Table 6. Individual and cumulative impact of the translation memory and online learning on the translation quality using the IT domain datasets

System	English-Estonian (BLEU)	English-Latvian (BLEU)
Baseline	26.80±0.17	26.42±0.23
Baseline + translation memory	28.87±0.20	28.97±0.26
(improvement)	(+2.07)	(+2.55)
OL+ + translation memory	31.45±0.20	38.59±0.31
(improvement)	(+2.58)	(+9.62)
Total improvement	**+4.65**	**+12.17**

in order to achieve the best results, it is beneficial to use both the translation memory and the online learning functionality.

8 Conclusion

In this paper, we described an online learning method for SMT systems based on the implementation by Bertoldi [3] and open source and freely available tools. We showed that the baseline implementation did not allow to improve SMT system quality due to sub-optimal tuning performance when adding the dynamic models to the SMT system. To address this issue, we devised a two-step tuning method, which, first, identifies good weights for the SMT system's static models and only then tunes the dynamic model weights in an online learning set-up.

Our experiments showed that the improved online learning method in combination with a translation memory allowed to increase IT domain SMT system quality from +4.65 (for English-Estonian) up to +12.17 (for English-Latvian) BLEU points. We also showed that although for narrow domain systems of very high quality (i.e., for systems of over 75 BLEU points) the online learning method did not show an improvement, the drop in quality is fairly minimal (just 0.55 BLEU points).

Finally, we analysed also the impact of text repetitiveness in the tuning dataset on the MT quality in online learning scenarios. The results showed that in order to achieve the highest MT quality, it is important for the tuning dataset to feature a level of text repetitiveness that matches the natural text repetitiveness of the data to be translated.

We believe that the findings of the paper will help other researchers and SMT system developers to successfully develop online learning systems that allow improving SMT system quality.

Acknowledgements. We would like to thank Tilde's Localization Department for the hard work they did to prepare the post-edited data analyses in this work. The research has been supported by the ICT Competence Centre (www.itkc.lv) within the project "2.2. Prototype of a Software and Hardware Platform for Integration of Machine Translation in Corporate Infrastructure" of EU Structural funds, ID n° 1.2.1.1/16/A/007.

References

1. Aziz, W., De Sousa, S.C., Specia, L.: Pet: a tool for post-editing and assessing machine translation. In: Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012), pp. 3982–3987 (2012)
2. Bentivogli, L., Bertoldi, N., Cettolo, M., Federico, M., Negri, M., Turchi, M.: On the evaluation of adaptive machine translation for human post-editing. IEEE/ACM Trans. Audio Speech Lang. Process. (TASLP) **24**(2), 388–399 (2016)
3. Bertoldi, N.: Dynamic models in Moses for online adaptation. Prague Bull. Math. Linguist. **101**, 7–28 (2014). https://doi.org/10.2478/pralin-2014-0001.Brought
4. Bertoldi, N., Cettolo, M., Federico, M.: Cache-based online adaptation for machine translation enhanced computer assisted translation. In: Proceedings of the XIV Machine Translation Summit, pp. 35–42 (2013)
5. Bertoldi, N., Haddow, B., Fouet, J.B.: Improved minimum error rate training in Moses. Prague Bull. Math. Linguist. **91**(1), 7–16 (2009)
6. Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., et al.: Findings of the 2017 conference on machine translation (wmt17). In: Proceedings of the Second Conference on Machine Translation, pp. 169–214 (2017)
7. Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Yepes, A.J., Koehn, P., Logacheva, V., Monz, C., et al.: Findings of the 2016 conference on machine translation. In: ACL 2016 First Conference on Machine Translation (WMT 2016), pp. 131–198. The Association for Computational Linguistics (2016)
8. Cettolo, M., Bertoldi, N., Federico, M.: The repetition rate of text as a predictor of the effectiveness of machine translation adaptation. In: Proceedings of the 11th Biennial Conference of the Association for Machine Translation in the Americas (AMTA 2014), pp. 166–179 (2014)
9. Denkowski, M., Lavie, A., Lacruz, I., Dyer, C.: Real time adaptive machine translation for post-editing with cdec and transcenter. In: Proceedings of the EACL 2014 Workshop on Humans and Computer-Assisted Translation, pp. 72–77 (2014)
10. Dyer, C., Chahuneau, V., Smith, N.A.: A simple, fast, and effective reparameterization of IBM model 2. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2013), Atlanta, USA, pp. 644–648, June 2013
11. Dyer, C., Weese, J., Setiawan, H., Lopez, A., Ture, F., Eidelman, V., Ganitkevitch, J., Blunsom, P., Resnik, P.: cdec: a decoder, alignment, and learning framework for finite-state and context-free translation models. In: Proceedings of the ACL 2010 System Demonstrations, pp. 7–12. Association for Computational Linguistics (2010)
12. Germann, U.: Dynamic phrase tables for machine translation in an interactive post-editing scenario. In: Proceedings of AMTA 2014 Workshop on Interactive and Adaptive Machine Translation, pp. 20–31 (2014)
13. Hasler, E., Haddow, B., Koehn, P.: Margin infused relaxed algorithm for moses. Prague Bull. Math. Linguist. **96**, 69–78 (2011)
14. Heafield, K.: KenLM: faster and smaller language model queries. In: Proceedings of the Sixth Workshop on Statistical Machine Translation, No. 2009, pp. 187–197. Association for Computational Linguistics (2011)

15. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, ACL 2007, Stroudsburg, PA, USA, pp. 177–180. Association for Computational Linguistics (2007). http://dl.acm.org/citation.cfm?id=1557769.1557821
16. Koehn, P., Och, F.J., Marcu, D.: Statistical phrase-based translation. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, vol. 1, pp. 48–54. Association for Computational Linguistics (2003)
17. Mathur, P., Cettolo, M.: Optimized MT online learning in computer assisted translation. In: IAMT 2014-AMTA 2014 Workshop on Interactive and Adaptive Machine Translation, pp. 32–41 (2014)
18. Mathur, P., Cettolo, M., Federico, M., Kessler, F.F.B.: Online learning approaches in computer assisted translation. In: WMT@ACL, pp. 301–308 (2013)
19. Microsoft: Translation and UI strings glossaries (2015)
20. Peris, Á., Casacuberta, F.: Online learning for effort reduction in interactive neural machine translation (2018). arXiv preprint: arXiv:1802.03594
21. Peris, A., Cebrián, L., Casacuberta, F.: Online learning for neural machine translation post-editing (2017). arXiv preprint: arXiv:1706.03196
22. Pinnis, M., Kalniņš, R., Skadiņš, R., Skadiņa, I.: What can we really learn from post-editing? In: Proceedings of the 12th Conference of the Association for Machine Translation in the Americas (AMTA 2016). MT Users, vol. 2, Austin, USA, pp. 86–91. Association for Machine Translation in the Americas (2016)
23. Skadiņa, I., Pinnis, M.: NMT or SMT: case study of a narrow-domain English-Latvian post-editing project. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing. Long Papers, vol. 1, pp. 373–383 (2017)
24. Tiedemann, J.: News from OPUS - a collection of multilingual parallel corpora with tools and interfaces. Recent Adv. Nat. Lang. Process. **5**, 237–248 (2009)
25. Turchi, M., Negri, M., Farajian, M.A., Federico, M.: Continuous learning from human post-edits for neural machine translation. Prague Bull. Math. Linguist. **108**(1), 233–244 (2017)
26. Vasiļjevs, A., Skadiņš, R., Tiedemann, J.: LetsMT!: a cloud-based platform for do-it-yourself machine translation. In: Proceedings of the ACL 2012 System Demonstrations, Jeju Island, Korea, pp. 43–48. Association for Computational Linguistics, July 2012