Measuring Immediate Adaptation Performance for Neural Machine Translation

Patrick Simianer, Joern Wuebker, John DeNero
Lilt
<given name>@lilt.com

Abstract

Incremental domain adaptation, in which a system learns from the correct output for each input immediately after making its prediction for that input, can dramatically improve system performance for interactive machine translation. Users of interactive systems are sensitive to the speed of adaptation and how often a system repeats mistakes, despite being corrected. Adaptation is most commonly assessed using corpus-level BLEU- or TER-derived metrics that do not explicitly take adaptation speed into account. We find that these metrics often do not capture immediate adaptation effects, such as zero-shot and one-shot learning of domain-specific lexical items. To this end, we propose new metrics that directly evaluate immediate adaptation performance for machine translation. We use these metrics to choose the most suitable adaptation method from a range of different adaptation techniques for neural machine translation systems.

1 Introduction

Incremental domain adaptation, or *online* adaptation, has been shown to improve statistical machine translation and especially neural machine translation (NMT) systems significantly (Turchi et al., 2017; Karimova et al., 2018) (*inter-alia*). The natural use case is a computer-aided translation (CAT) scenario, where a user and a machine translation system collaborate to translate a document. Each user translation is immediately used as a new training example to adapt the machine translation system to the specific document.

Adaptation techniques for MT are typically evaluated by their corpus translation quality, but such evaluations may not capture prominent aspects of the user experience in a collaborative translation scenario. This paper focuses on directly measuring the speed of lexical acquisition for in-domain vocabulary. To that end, we propose three related metrics that are designed to reflect the responsiveness of adaptation.

An ideal system would immediately acquire in-domain lexical items upon observing their translations. Moreover, one might expect a neural system to generalize from one corrected translation to related terms. Once a user translates "bank" to German "Bank" (*institution*) instead of "Ufer" (*shore*) in a document, the system should also correctly translate "banks" to "Banken" instead of "Ufer" (the plural is identical to the singular in German) in future sentences. We measure both one-shot vocabulary acquisition for terms that have appeared once in a previous target sentence, as well as zero-shot vocabulary acquisition for terms that have not previously appeared.

Our experimental evaluation shows some surprising results. Methods that appear to have comparable performance using corpus quality metrics such as BLEU can differ substantially in zero-shot and one-shot vocabulary acquisition. In addition, we find that *fine-tuning* a neural model tends to improve one-shot vocabulary recall while *degrading* zero-shot vocabulary recall.

We evaluate several adaptation techniques on a range of online adaptation datasets. Fine tuning applied to all parameters in the NMT model maximizes one-shot acquisition, but shows a worrisome degradation in zero-shot recall. By contrast, fine tuning with group lasso regularization, a technique recently proposed to improve the space efficiency of adapted models (Wuebker et al., 2018), achieves an appealing balance of zero-shot and one-shot vocabulary acquisition as well as high corpus-level translation quality.

2 Measuring Immediate Adaptation

2.1 Motivation

For interactive, adaptive machine translation systems, *perceived* adaptation performance is a crucial property: An error in the machine translation output which needs to be corrected multiple times can cause frustration, and thus may compromise acceptance of the MT system by human users. A class of errors that are particularly salient are lexical choice errors for domain-specific lexical items. In the extreme, NMT systems using subword modeling (Sennrich et al., 2015) can generate "hallucinated" words—words that do not exist in the target language—which are especially irritating for users (Lee et al., 2018; Koehn and Knowles, 2017). Users of adaptive MT have a reasonable expectation that in-domain vocabulary will be translated correctly after the translation of a term or some related term has been corrected manually.

Arguably, more subtle errors, referring to syntax, word order or more general semantics are less of a focus for immediate adaptation, as these types of errors are also harder to pinpoint and thus to evaluate[1] (Bentivogli et al., 2016). Traditional metrics for evaluating machine translation outputs, e.g. BLEU and TER, in essence try to measure the similarity of a hypothesized translation to one or more reference translations, taking the full string into account.

Due to significant improvements in MT quality with neural models (Bentivogli et al., 2016) (*inter-alia*), more specialized metrics, evaluating certain desired behaviors of systems become more useful for specific tasks. For example, Wuebker et al. (2016) show, that NMT models, while being better in most respects, still fall short in the handling of content words in comparison with phrase-based MT. This observation is also supported by Bentivogli et al. (2016), who show smaller gains for NMT for translation of nouns, an important category of content words.

Another reason to isolate vocabulary acquisition as an evaluation criterion is that interactive translation often employs local adaptation via prefix-decoding (Knowles and Koehn, 2016; Wuebker et al., 2016), which can allow the system to recover syntactic structure or resolve local ambiguities when given a prefix, but may still suffer from poor handling of unknown or domain-specific vocabulary.

In this work, we therefore focus on translation performance with respect to content words, setting word order and other aspects aside.

2.2 Metrics

We propose three metrics: one to directly measure one-shot vocabulary acquisition, one to measure zero-shot vocabulary acquisition, and one to measure both. In all three, we measure the recall of target-language content words so that the metrics can be computed automatically by comparing translation hypotheses to reference translations without the use of models or word alignments[2].

We define content words as those words that are not included in a fixed stopword list, as used for example in query simplification for information retrieval. Such lists are typically compiled manually and are available for many languages.[3] For western languages, content words are mostly nouns, main verbs, adjectives or adverbs.

For the i-th pair of source sentence and reference translation, $i = 1, \ldots, |\mathcal{G}|$, of an ordered test corpus $\mathcal{G}$, we define two sets $\mathcal{R}_{0,i}$ and $\mathcal{R}_{1,i}$ that are a subset of the whole set of unique[4] content words (i.e. types) of the reference translation for i. $\mathcal{R}_{0,i}$ includes a word if its first occurrence in the test set is in the i-th reference of $\mathcal{G}$, and $\mathcal{R}_{1,i}$ if its second occurrence in the test set is in the i-th reference of $\mathcal{G}$. The union $\mathcal{R}_{0,i} \cup \mathcal{R}_{1,i}$ includes content words occurring for either the first or second time.

To measure zero-shot adaptation in a given hypothesis $\mathcal{H}_i$, also represented as a set of its content words, we propose to evaluate the number of word types that were immediately translated correctly:

$$\text{R0} = \frac{|\mathcal{H}_i \cap \mathcal{R}_{0,i}|}{|\mathcal{R}_{0,i}|}.$$

To measure one-shot adaptation, where the system correctly produces a content word after ob-

[1] Some practitioners observed that these subtle errors become harder to spot due the improved fluency of NMT systems (Burchardt, 2017).

[2] In each of the data sets considered in this work, the average number of occurrences of content words ranges between 1.01 and 1.11 per sentence. We find this sufficiently close to 1 to evaluate in a bag-of-words fashion and not consider alignments.

[3] For German we used the list available here: https://github.com/stopwords-iso.

[4] All proposed metrics operate on the set-level, without clipping (Papineni et al., 2002) or alignment (Banerjee and Lavie, 2005; Kothur et al., 2018), as we have found this simplification effective.

	Reference	Hypothesis	R0	R1	R0+1
1.	The [dog] [bites] the [lady]	A [terrier] [bites] the [person]	1/3	0/0	1/3
2.	The [man] [bites] the [dog]	The [dog] [bites] the [man]	1/1	2/2	3/3
		Total	2/4	2/2	4/6

Figure 1: Example for calculating R0, R1, and R0+1 on a corpus of two sentences. Content words are written in brackets, the corpus-level score is given below the per-segment scores. In the example, the denominator for R1 is 2 due to the two repeated words *dog* and *bites* in the reference.

serving it exactly once, we propose:

$$\text{R1} = \frac{|\mathcal{H}_i \cap \mathcal{R}_{1,i}|}{|\mathcal{R}_{1,i}|}.$$

This principle can be extended to define metrics Rk, $k > 1$ to allow more "slack" in the adaptation, but we leave that investigation to future work.

Finally, we define a metric that measures both zero- and one-shot adaptation:

$$\text{R0+1} = \frac{|\mathcal{H}_i \cap [\mathcal{R}_{0,i} \cup \mathcal{R}_{1,i}]\,|}{|\mathcal{R}_{0,i} \cup \mathcal{R}_{1,i}|}.$$

All metrics can either be calculated for single sentences as described above, or for a full test corpus by summing over all sentences, e.g. for R0:

$$\frac{\sum_{i=1}^{|\mathcal{G}|} |\mathcal{H}_i \cap \mathcal{R}_{0,i}|}{\sum_{i=1}^{|\mathcal{G}|} |\mathcal{R}_{0,i}|}.$$

Figure 1 gives an example calculation of all three metrics across a two-sentence corpus.

3 Related Work

An important line of related work is concerned with estimating the *potential* adaptability of a system given a source text only, the so-called repetition rate (Cettolo et al., 2014). The metric is inspired by BLEU, and uses a sliding window over the source text to count singleton N-grams.

The modus operandi for our metrics is most similar to HTER (Snover et al., 2006), since we are also assuming a single, targeted reference translation[5] for evaluation.

The introduction of NMT brought more aspects of translation quality evaluation into focus, such as discourse-level evaluation (Bawden et al., 2017), or very fine-grained evaluation of specific aspects of the translations (Bentivogli et al., 2016), highlighting the differences between phrase-based and NMT systems.

[5]A reference translation which was produced from post-editing output of the to-be-evaluated MT system.

Online adaptation for (neural) machine translation has been thoroughly explored using BLEU (Turchi et al., 2017), simulated keystroke and mouse action ratio (Barrachina et al., 2009) for effort estimation (Peris and Casacuberta, 2018), word prediction accuracy (Wuebker et al., 2016), and user studies (Denkowski et al., 2014; Karimova et al., 2018) (all *inter-alia*). In (Simianer et al., 2016) immediate adaptation for hierarchical phrase-based MT is specifically investigated, but they also evaluate their systems using human-targeted BLEU and TER.

Regularization for segment-wise continued training in NMT has been explored by Khayrallah et al. (2018) by means of knowledge distillation, and with the group lasso by Wuebker et al. (2018), as used in this paper.

Most relevant to our work, in the context of document-level adaptation, Kothur et al. (2018) calculate accuracy for novel words based on an automatic word alignment. However, they do not focus on zero- and one-shot matches, but instead aggregate counts over the full corpus.

4 Online Adaptation

NMT systems can be readily adapted by fine-tuning (also called continued training) with the same cross-entropy loss ($\mathcal{L}$) as used for training the parameters of the baseline system, which also serves as the starting point for adaptation (Luong and Manning, 2015). Following Turchi et al. (2017), we perform learning from each example i using (stochastic) gradient descent, using the current source x_i and reference translation y_i as a batch of size 1:

$$\theta_i \leftarrow \theta_{i-1} - \gamma \nabla \mathcal{L}(\theta_{i-1}, x_i, y_i). \quad (1)$$

Evaluation is carried out using simulated post-editing (Hardt and Elming, 2010), first translating the source using the model with parameters θ_{i-1}, before performing the update described

above with the now revealed reference translation. The machine translation system effectively only trains for a single iteration for any given data set.

The naïve approach, updating all parameters θ of the NMT model, while being effective, can be infeasible in certain settings[6], since tens of millions of parameters are updated depending on the respective model. While some areas of a typical NMT model can be stored in a sparse fashion without loss (source- and target embeddings), large parts of the model cannot. We denote this type of adaptation as *full*.

A light-weight alternative to adaptation of the full parameter set is to introduce a second bias term in the final output layer of the NMT model, which is trained in isolation, freezing the rest of the model (Michel and Neubig, 2018). This merely introduces a vector in the size of the output vocabulary. This method is referred to as *bias*.

Another alternative is freezing parts of the model (Thompson et al., 2018), for example determining a subset of parameters by performance on a held-out set (Wuebker et al., 2018). In our experiments we use two systems using this method, *fixed* and *top*, the former being a pre-determined fixed selection of parameters, and the latter being the topmost encoder and decoder layers in the *Transformer* NMT model (Vaswani et al., 2017).

Finally, a data-driven alternative to the fixed freezing method was introduced to NMT by Wuebker et al. (2018), implementing tensor-wise ℓ_1/ℓ_2 group lasso regularization, allowing the learning procedure to select a fixed number of parameters after each update. This setup is referred to as *lasso*.

5 Experiments

5.1 Neural Machine Translation Systems

We adapt an English→German NMT system based on the Transformer architecture trained with an in-house NMT framework on about 100M bilingual sentence pairs. The model has six layers in the encoder, three layers in the decoder, each with eight attention heads with dimensionality 256, distinct input and output embeddings, and vocabulary sizes of around 40,000. The vocabularies are generated with byte-pair encoding (Sennrich et al., 2015). For adaptation we use a learning rate γ of 10^{-2} (for the *bias* adaptation a learning rate of 1.0 is used), no dropout, and no label-smoothing. We use a tensor-wise ℓ_2 normalization to 1.0 for all gradients (gradient clipping). Updates for a sentence pair are repeated until the perplexity on that sentence pair is ≤ 2.0, for a maximum of three repetitions. The *fixed* adaptation scheme, which involves selecting a subset of parameters on held-out data following Wuebker et al. (2018), uses about two million parameters excluding all embedding matrices, in addition to potentially the full source embeddings, but in practice this is limited to about 1M parameters. The *top* scheme only adapts the top layers for both encoder and decoder. For the *lasso* adaptation, we allow 1M parameters excluding the embeddings, for which we allow 1M parameters in total selected from all embedding matrices. This scheme also always includes the previously described second bias term in the final output layer.

[6]For example in setups where a large number of these adapted models need to be stored and transferred.

Method	BLEU	SBLEU	TER
baseline	40.3	49.3	45.2
bias	40.4	49.5	45.0
full	47.0	**55.9**	44.0
lasso	46.3	54.3	42.6
fixed	**47.1**	55.5	**41.0**
top	43.2	54.0	49.5

Table 1: Results on the *Autodesk* test set for traditional MT quality metrics. *SBLEU* refers to an average of sentence-wise BLEU scores as described by Nakov et al. (2012). The best result in each column is denoted with bold font.

Since the proposed metrics operate on words, the machine translation outputs are first converted to full-form words using *sentencepiece* (Kudo and Richardson, 2018), then tokenized and truecased with the tokenizer and truecaser distributed with the *Moses* toolkit (Koehn et al., 2007).

5.2 Results

Tables 1 and 2 show the performance of different adaptation techniques on the *Autodesk* dataset (Zhechev, 2012), a public post-editing software domain dataset for which incremental adaptation is known to provide large gains for corpus-level metrics. BLEU, sentence BLEU, and TER scores (Table 1) are similar for *full* adaptation, sparse adaptation with group *lasso*, and adaptation of a *fixed* subset of parameters. However (in Table 2),

Method	R0	R1	R0+1
baseline	39.3	44.9	41.0
bias	39.3	45.3	41.1
full	35.8	**55.0**	41.6
lasso	**40.3**	48.6	**42.8**
fixed	35.8	52.3	40.8
top	35.6	50.3	40.0

Table 2: Results on the *Autodesk* test set for the proposed metrics R0, R1, and R0+1.

lasso substantially outperforms the other methods in zero-shot (R0), and combined zero- and one-shot recall of content words (R0+1).

Zero-shot recall is considerably degraded relative to the non-adapted baseline for both *full* and adaptation of a fixed subset of tensors (*fixed* and *top*). That is, terms never observed before during online training are translated correctly less often than they would be with an unadapted system, despite the data set's consistent domain. These approaches trade off long-term gains in BLEU and high one-shot recall for low zero-shot recall, which could be frustrating for users who may perceive the degradation in quality for terms appearing for the first time in a document. The *lasso* technique is the only one that shows an improvement in R0 over the baseline. However, *lasso* has considerably lower one-shot recall compared to the other adaptation methods, implying that it often must observe a translated term more than once to acquire it.

Appendix A shows similar experiments for several other datasets.

5.3 Analysis

For a better understanding of the results described in the previous section, we conduct an analysis varying the units of the proposed metrics, while focusing on *full* and *lasso* adaptation.

For the first variant, only truly novel words are taken into account, i.e. words in the test set that do not appear in the training data. Results for these experiments are depicted in Table 3. It is apparent that the findings of Table 2 are confirmed, and that relative differences are amplified. This can be explained by the reduced number of total occurrences considered, which is only 310 words in this data set. It is also important to note that all of these words are made up of known subwords[7], since our NMT system does not include a copying mechanism and is thus constrained to the target vocabulary.

Method	R0	R1	R0+1
baseline	27.1	40.7	29.9
full	26.1	**63.0**	33.8
lasso	**31.9**	53.1	**36.3**

Table 3: Results on *Autodesk* data calculating the metrics only for truly novel content words, i.e. ones that do not occur in the training data.

Method	R0	R1	R0+1
baseline	**44.1**	48.1	45.5
full	40.4	**54.6**	45.4
lasso	43.7	51.7	**46.5**

Table 4: Results on *Autodesk* data calculating the metrics with subwords.

Further results using the raw subword output[8] of the MT systems are depicted in Table 4: R0 for the *lasso* method is degraded only slightly below the baseline (-1%, compared to +2% for the regular metric), the findings for R1 and R0+1 remain the same as observed before. Compared to the results for novel words this indicates that the improvement in terms of R0 for *lasso* mostly come from learning new combinations of subwords.

A discussion of the adaptation behavior over time, with exemplified differences between the metrics, can be found in Appendix B.

6 Conclusions

To summarize: In some cases, the strong gains in corpus-level translation quality achieved by fine tuning an NMT model come at the expense of zero-shot recall of content words. This concerning impact of adaptation could affect practical user experience. Existing regularization methods mitigate this effect to some degree, but there may be more effective techniques for immediate adaptation that have yet to be developed.

The proposed metrics R0, R1, and R0+1 are useful for measuring immediate adaptation performance, which is crucial in adaptive CAT systems.

[7] The test set does not contain any unknown characters.

[8] Note that this includes all tokens, not just parts of content words.

References

Satanjeev Banerjee and Alon Lavie. 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*, pages 65–72.

Sergio Barrachina, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, et al. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28.

Rachel Bawden, Rico Sennrich, Alexandra Birch, and Barry Haddow. 2017. Evaluating discourse phenomena in neural machine translation. *arXiv preprint arXiv:1711.00513*.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. Neural versus phrase-based machine translation quality: a case study. *arXiv preprint arXiv:1608.04631*.

Aljoscha Burchardt. 2017. Comparing errors: Neural MT vs. traditional phrase-based and rule-based MT. https://www.gala-global.org/publications/comparing-errors-neural-mt-vs-traditional-phrase-based-and-rule-based-mt.

Mauro Cettolo, Nicola Bertoldi, and Marcello Federico. 2014. The repetition rate of text as a predictor of the effectiveness of machine translation adaptation. In *Proceedings of AMTA*, pages 166–179.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. WIT^3: Web inventory of transcribed and translated talks. In *Proceedings of EAMT*, pages 261–268.

Michael Denkowski, Alon Lavie, Isabel Lacruz, and Chris Dyer. 2014. 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*, pages 72–77.

Daniel Hardt and Jakob Elming. 2010. Incremental retraining for post-editing SMT. In *Proceedings of AMTA*, volume 122.

Sariya Karimova, Patrick Simianer, and Stefan Riezler. 2018. A user-study on online adaptation of neural machine translation to human post-edits. *Machine Translation*, 32(4):309–324.

Huda Khayrallah, Brian Thompson, Kevin Duh, and Philipp Koehn. 2018. Regularized training objective for continued training for domain adaptation in neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 36–44.

Rebecca Knowles and Philipp Koehn. 2016. Neural interactive translation prediction. In *Proceedings of AMTA*, pages 107–120.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL*, pages 177–180.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872*.

Sachith Sri Ram Kothur, Rebecca Knowles, and Philipp Koehn. 2018. Document-level adaptation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 64–73.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in neural machine translation. https://openreview.net/forum?id=SkxJ-309FQ.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of IWSLT*, pages 76–79.

Paul Michel and Graham Neubig. 2018. Extreme adaptation for personalized neural machine translation. *arXiv preprint arXiv:1805.01817*.

Preslav Nakov, Francisco Guzman, and Stephan Vogel. 2012. Optimizing for sentence-level BLEU+1 yields short translations. *Proceedings of COLING*, pages 1979–1994.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318.

Álvaro Peris and Francisco Casacuberta. 2018. Online learning for effort reduction in interactive neural machine translation. *arXiv preprint arXiv:1802.03594*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Patrick Simianer, Sariya Karimova, and Stefan Riezler. 2016. A post-editing interface for immediate adaptation in statistical machine translation. In *Proceedings of COLING*, pages 16–20, Osaka, Japan.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of AMTA*, volume 200.

Brian Thompson, Huda Khayrallah, Antonios Anastasopoulos, Arya McCarthy, Kevin Duh, Rebecca Marvin, Paul McNamee, Jeremy Gwinnup, Tim Anderson, and Philipp Koehn. 2018. Freezing subnetworks to analyze domain adaptation in neural machine translation. In *Proceedings of WMT*, pages 124–132.

Marco Turchi, Matteo Negri, M Amin Farajian, and Marcello Federico. 2017. Continuous learning from human post-edits for neural machine translation. *The Prague Bulletin of Mathematical Linguistics*, 108(1):233–244.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NIPS*, pages 5998–6008.

Katharina Wäschle and Stefan Riezler. 2012. Analyzing parallelism and domain similarities in the MAREC patent corpus. *Multidisciplinary Information Retrieval*, pages 12–27.

Joern Wuebker, Spence Green, John DeNero, Sasa Hasan, and Minh-Thang Luong. 2016. Models and inference for prefix-constrained machine translation. In *Proceedings of ACL*, volume 1, pages 66–75.

Joern Wuebker, Patrick Simianer, and John DeNero. 2018. Compact personalized models for neural machine translation. In *Proceedings of EMNLP*, pages 881–886.

Ventsislav Zhechev. 2012. Machine translation infrastructure and post-editing performance at autodesk. In *Proceedings of WPTP*, pages 87–96.

A Additional Results

Table 5 contains results for additional English→German datasets, namely patents (Wäschle and Riezler, 2012) (*Patent*), transcribed public speeches (Cettolo et al., 2012) (*TED*), and two proprietary user data sets, one from the financial domain (*User 1*) and the other being technical documentation (*User 2*). The same pattern is observed in almost all cases: *lasso* outperforms the other adaptation techniques in zero-shot recall (R0) and combined recall (R0+1), while *full* has the highest one-shot recall (R1) on two out of five test sets, being close runner-up to *lasso* on all others. Overall however, we observe that zero-shot recall R0 is degraded by adaptation, while one-shot recall is improved. We also find that adaptation with the light-weight *bias* method often does not deviate much from the baseline. In contrast, the results for the traditional MT metrics are predominantly positive. For adaptation, the *lasso* method provides the best tradeoff in terms of performance throughout the considered metrics.

B Learning Curves

We are also interested in the behavior of the adaptation methods over time. To this end, in Figure 2, we plot the difference in cumulative scores[9] of two adapted systems (*full* and *lasso*) to the baseline for the proposed metrics as well as the BLEU score.

As evident from comparing the curves for BLEU and R0, the BLEU score and the proposed metric give disparate signals for this data. Specifically, there are two distinct dips in the curves for R0 (as well as R0+1) and BLEU:

1. The degradation in R0 around segment 800 is due to significant noise in segment 774, which has a strong impact on the adapted systems, while the baseline system is not affected. The *full* system's score drops by about 50% at segment 775 (i.e. after adaptation) relative to the cumulative score difference at the previous segment and never recovers after that.

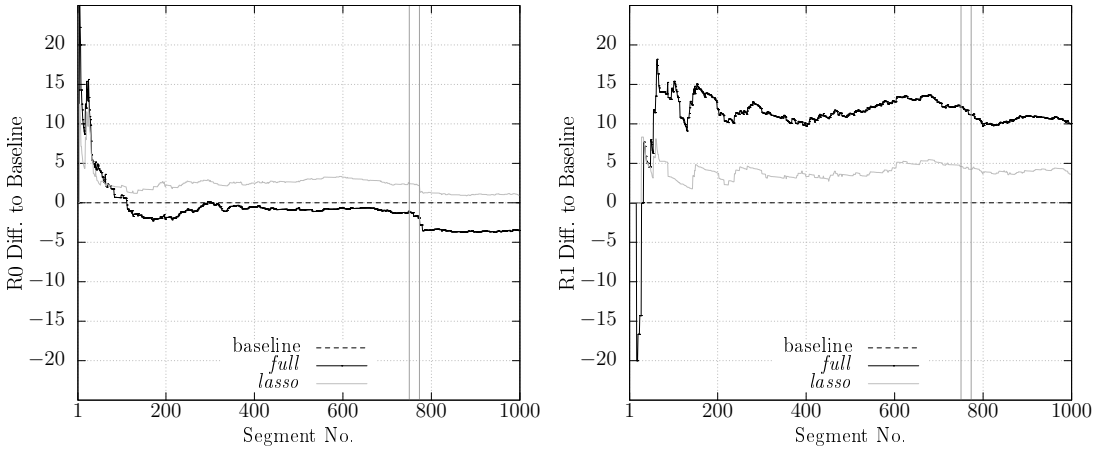
2. The dip in the BLEU score at segment 752, observable for both adapted systems, depicting a relative degradation of about 10%, is due to a pathological repetition of a single character in the output of the adapted MT systems for this segment, which has a large impact on the score.

The dip observed with R0 is also noticeable in BLEU, but much less pronounced at 4% relative for *full* and 2% relative for *lasso*. The dip in BLEU on the other hand is not noticeable with R0, R1, or R0+1.

[9] For each sentence i in the data set, the metrics for all systems are calculated up to the ith sentence. The difference for the adapted systems is then calculated by subtracting the baseline score.

User 1	BLEU	SBLEU	TER	R0+1	R0	R1
baseline	35.7	55.2	52.4	44.3	42.8	50.3
bias	8	6	-4	-5	-5	-4
full	36	18	-22	-4	-7	6
lasso	38	18	-23	1	-1	8
fixed	34	18	-22	-6	-9	4
top	29	16	-17	-5	-8	4
User 2	BLEU	SBLEU	TER	R0+1	R0	R1
baseline	35.5	56.2	51.0	43.6	41.0	51.2
bias	0	0	0	0	0	-1
full	0	5	5	-3	-5	4
lasso	6	6	-6	2	0	7
fixed	-5	4	13	-4	-7	1
top	-3	3	4	-5	-7	-2
Autodesk	BLEU	SBLEU	TER	R0+1	R0	R1
baseline	40.3	49.3	45.2	41.0	39.3	44.9
bias	0	0	0	0	0	1
full	17	13	-3	1	-9	22
lasso	15	10	-6	4	3	8
fixed	17	13	-9	0	-9	16
top	7	10	10	-2	-9	12
TED	BLEU	SBLEU	TER	R0+1	R0	R1
baseline	25.9	56.0	54.2	42.6	39.5	53.2
bias	1	0	0	0	0	0
full	0	1	1	-3	-6	3
lasso	4	2	-2	-1	-3	4
fixed	-3	0	4	-4	-7	2
top	-6	0	9	-2	-5	5
Patent	BLEU	SBLEU	TER	R0+1	R0	R1
baseline	53.5	62.1	31.7	51.8	49.7	57.3
bias	2	1	-2	0	0	0
full	3	2	-2	-2	-5	7
lasso	4	2	-4	0	-2	5
fixed	2	1	1	-4	-7	4
top	2	1	-1	-3	-5	2

Table 5: BLEU, sentence-wise BLEU, TER, R0+1, R0, and R1 metrics for a number of data sets, comparing different adaptation methods as described in Section 4. Baseline results are given as absolute scores, results for adaptation are given as relative differences. Best viewed in color.



(a) R0 (b) R1

(c) R0+1 (d) BLEU

Figure 2: Differences in cumulative scores for R0 (top left), R1 (top right), R0+1 (bottom left), and BLEU (bottom right) to the baseline system on the *Autodesk* test set for *full* and *lasso* adaptation. The peculiarities discussed in the running text are marked by solid vertical lines (at $x = 751$ and $x = 774$).