Robust Tuning Datasets for Statistical Machine Translation

Preslav Nakov and **Stephan Vogel**
ALT Research Group
Qatar Computing Research Institute, HBKU
{pnakov, svogel}@hbku.edu.qa

Abstract

We explore the idea of automatically crafting a tuning dataset for Statistical Machine Translation (SMT) that makes the hyper-parameters of the SMT system more robust with respect to some specific deficiencies of the parameter tuning algorithms. This is an under-explored research direction, which can allow better parameter tuning. In this paper, we achieve this goal by selecting a subset of the available sentence pairs, which are more suitable for specific combinations of optimizers, objective functions, and evaluation measures. We demonstrate the potential of the idea with the pairwise ranking optimization (PRO) optimizer, which is known to yield too short translations. We show that the learning problem can be alleviated by tuning on a subset of the development set, selected based on sentence length. In particular, using the longest 50% of the tuning sentences, we achieve two-fold tuning speedup, and improvements in BLEU score that rival those of alternatives, which fix BLEU+1's smoothing instead.

1 Introduction

Modern Statistical Machine Translation (SMT) systems have several, somewhat independent, components that work together to generate a good translation, and are typically combined in a log-linear framework, where the language model, the translation model, the reordering model, etc., contribute to the hypothesis score with different weights. It is now standard to learn these weights discriminatively, from a development dataset, e.g., by optimizing BLEU (Papineni et al., 2002) or some other measure directly.

A tuned system can often yield very significant improvements in terms of translation quality compared to a system that uses standard, untuned default parameters. Thus, a lot of research attention in SMT has been devoted to designing different algorithms for parameter optimization. For years, it was typical to use minimum error rate training, or MERT (Och, 2003), which works quite well when the number of parameters is small. As the number of parameters has grown, rivaling optimizers such as MIRA (Watanabe et al., 2007; Chiang et al., 2008) and PRO (Hopkins and May, 2011) have been developed, as well as various variations thereof (Bazrafshan et al., 2012; Cherry and Foster, 2012; Gimpel and Smith, 2012).

These optimization algorithms have focused on learning from a given fixed dataset, relying on the standard machine learning assumption that the training and the development data come from the same distribution as the test data, e.g., in terms of domain, coverage, genre, length, etc. In practical terms, this is especially important for the development/tuning data, but with standard datasets, there is often no way to guarantee this, and many researchers have determined empirically the most suitable tuning dataset by observing the translation score on the test dataset. Yet, the choice of tuning dataset can considerably affect the results; for example, Zheng et al. (2010) report variation across different standard NIST tuning MTxx datasets of over six BLEU points for Chinese-English SMT when testing on the NIST MT08 test dataset.

Given a *reasonable* tuning set, i.e., one that is really coming from the same distribution as the test dataset, a *good* optimization algorithm should be able to learn to produce optimal weights. Yet, the choice of optimization objective can yield dramatically different translations since different algorithms might need to stress some aspects of the tuning dataset and downplay others.

Proceedings of Recent Advances in Natural Language Processing, pages 543–550,
Varna, Bulgaria, Sep 4–6 2017.
https://doi.org/10.26615/978-954-452-049-6_071

One reason for this is that different optimizers interact differently with different objectives. For example, we have previously shown that sentence-level optimizers yield too short translations when optimizing BLEU+1 (Nakov et al., 2012).

In this paper, we advocate the idea of automatically crafting a tuning dataset that makes tuning parameters *less* susceptible to the deficiencies of the learning algorithms. More specifically, in order to bridge this gap, we propose to customize the tuning dataset by selecting a subset of the available sentence pairs, taking into account the target domain and the peculiarities of the optimization algorithm, objective function, and evaluation measure used. This is important because it brings us a step closer to robust learning, instead of simply overfitting the tuning dataset.

Below, we focus specifically on sentence length as a selection criteria. We chose length because it can have consequences on how translations are scored w.r.t. to metrics like BLEU, and besides, it is a known issue for PRO. Still, to the best of our knowledge, the interaction between the optimizer, the optimization objective, the evaluation measure, and the development dataset has been largely neglected so far. For instance, we show that the problem of short translations when tuning with PRO (Hopkins and May, 2011) is worsened when tuning on short sentence pairs, and alleviated when emphasizing the longer tuning sentences.

Tuning set crafting can be done in various ways, e.g., by removing examples with sub-optimal characteristics (e.g., short sentences) or by oversampling the ones with desired characteristics (e.g., longer sentences). Here we focus on selection from a single tuning set because it is applicable to different datasets. We show that significant performance gains are possible with PRO when selecting just a subset of the tuning dataset based on length. Our objective here is to draw the attention of the research community to the possibilities that dataset customization through subset selection can offer for different experimental conditions. We believe that this is a very promising, yet largely underexplored research direction.

Naturally, one could also try to select data from elsewhere and build a completely custom dataset, e.g., by selecting sentences from the training dataset. However, this is not possible in case of multiple references (the training bi-text has only one reference). One could also try to select/fuse from different available tuning datasets, but it is rare to have multiple tuning datasets.

It has also been observed that having multiple references in the tuning dataset can yield more accurate parameter estimates and thus better test translation scores (Madnani et al., 2008). Thus, adding a few human translations, could significantly boost translation quality. Since this is costly, some researchers have resorted to using automatically generated references, with modest performance gains.

The remainder of the paper is organized as follows: Section 2 introduces related work, Section 3 describes the method, Section 4 presents the experimental setup, Section 5 discusses the evaluation results, and Section 6 provides deeper analysis and further discussion. Finally, Section 7 concludes with possible directions for future work.

2 Related Work

Tuning the parameters of a log-linear model for SMT is an active area of research. The typical way to do this is to use minimum error rate training, or MERT, (Och, 2003), which optimizes the standard dataset-level BLEU directly.

Recently, there has been a surge in new optimization techniques for SMT. Most notably, this includes the margin-infused relaxed algorithm or MIRA (Watanabe et al., 2007; Chiang et al., 2008, 2009), which is an on-line sentence-level perceptron-like passive-aggressive optimizer, and pairwise ranking optimization or PRO (Hopkins and May, 2011), which operates in batch mode and sees tuning as ranking.

A number of improved versions thereof have been proposed including a batch version of MIRA (Cherry and Foster, 2012) and a linear regression version of PRO (Bazrafshan et al., 2012). Another recent optimizer is Rampeon (Gimpel and Smith, 2012). We refer the interested reader to three recent overviews on parameter optimization for SMT: (McAllester and Keshet, 2011; Cherry and Foster, 2012; Gimpel and Smith, 2012).

With the emergence of new optimization techniques, there have been also studies that compare stability between MIRA–MERT (Chiang et al., 2008, 2009; Cherry and Foster, 2012), PRO–MERT (Hopkins and May, 2011), MIRA–PRO–MERT (Cherry and Foster, 2012; Gimpel and Smith, 2012; Nakov et al., 2012).

More relevant to the present work, there has been some interest in analyzing how different optimizers interact with specific metrics. For example, pathological verbosity was reported when tuning MERT on recall-oriented metrics such as METEOR (Lavie and Denkowski, 2009; Denkowski and Lavie, 2011), large variance was observed with MIRA (Simianer et al., 2012), and *monsters* were found when using PRO with too long tuning sentences (Nakov et al., 2013b). In previous work, we also found that MERT learns verbosity, while PRO learns length (Guzmán et al., 2015b).

It has been also observed that having multiple references for the tuning dataset can yield better test-time translation performance (Madnani et al., 2007). Thus, adding a few extra human reference translations could significantly boost translation quality. Since this is costly, some researchers have resorted to using automatically generated references,[1] via paraphrasing (Madnani et al., 2008) and back-translation (Dyer et al., 2011), with modest performance gains.

There has been also work on tuning data selection and/or fusion in the special case when multiple versions of the source sentence are available (Nakov et al., 2013a). This is a fairly rare situation for such approaches to be broadly applicable.

Most relevant to our work, there were efforts to build tuning datasets using information retrieval (Zheng et al., 2010; Tamchyna et al., 2012), text clustering (Li et al., 2010), and sentence-length based features (Guzmán et al., 2012). To avoid data sparseness, most of these approaches require a larger pool of data, which they typically select from the training bi-text, thus reducing the amount of data available for model training. This makes such approaches inapplicable in multi-reference testset contexts since the bi-text only has one translation per source sentence. Moreover, the selection is typically done based on the actual test input, which is not known a priori in a realistic SMT setup, e.g., in online translation.

In contrast, we aim to produce customized datasets that are less susceptible to that, and are suitable for specific combinations of optimizers, objective functions, and evaluation measures. This can yield better parameter estimation, while using less data and more efficient tuning with faster iterations and a smaller computational footprint.

[1]Paraphrasing was also applied to the training bi-text (Nakov, 2008; Nakov and Ng, 2009) and to the phrase table (Callison-Burch et al., 2006).

3 Method

Below we present one particular example of tuning dataset customization in order to illustrate the potential of the idea.

In previous work, we have shown that the PRO optimizer yields SMT parameters that yield test-time translations that are shorter than they should be. We have addressed this by changing the objective function, sentence-level BLEU+1, and we have proposed to replace it with one with better smoothing (Nakov et al., 2012). Here we propose an alternative solution, which customizes the tuning dataset by selecting a subset of higher average length.

Observe that, if the reason for PRO yielding too short translations is the add-one smoothing in BLEU+1, this should affect shorter sentences to a greater extent, since the effect of the smoothing is bigger for them. I.e., we should expect that, when tuning with PRO, the translations of short sentences should get relatively shorter translations than those of long sentences. This means that we should expect to get longer translations if we tune on longer sentences, i.e., if we customize the tuning dataset, which can be done, e.g., (*a*) by excluding some of the short sentences or (*b*) by oversampling some of the long sentences. We will explore approach (*a*) below: in particular, we will exclude half of the sentences, keeping the longest 50% only.

4 Experimental Setup

We experimented with Arabic-to-English SMT, training on the Arabic-English data that was made available for the NIST 2012 OpenMT Evaluation.[2] We used all training data except for the UN corpus, we tuned on MT06 (and subsets thereof), and we tested on MT09, which have four English reference translations.

We trained a phrase-based SMT model (Koehn et al., 2003) as implemented in the Moses toolkit (Koehn et al., 2007). We tokenized and truecased the English side of the training/development/testing bitexts, and the monolingual data for language modeling using the standard tokenizer of Moses. We segmented the words on the Arabic side of all bitexts using the MADA ATB segmentation scheme (Roth et al., 2008).

[2]www.nist.gov/itl/iad/mig/openmt12.cfm

	Tuning	BLEU	BP
1	BP-smooth=1, grounded	47.61	0.991
2	BP-smooth=1	47.52	0.984
3	top50	**47.47**	**0.980**
4	mid50	47.44	0.977
5	rand50	47.43	0.978
6	low50	46.38	0.961
7	full	**47.18**	**0.972**

Table 1: Multi-reference PRO experiments: test-set BLEU and BP when tuning on different length-based subsets of the tuning dataset (lines 3-6). For comparison, we also show the results when tuning on the full tuning dataset (line 7), as well as what the PRO-fixes proposed in (Nakov et al., 2012) would achieve when tuning on the full dataset (lines 1-2).

	Tuning	BLEU	BP
1	BP-smooth=1, grounded	29.68	0.979
2	BP-smooth=1	29.43	0.962
3	top50	**29.51**	**0.969**
4	mid50	29.11	0.950
5	rand50	28.96	0.941
6	low50	27.44	0.894
7	full	**28.88**	**0.934**

Table 2: Single-reference PRO experiments: test-set BLEU and BP when tuning on different length-based subsets of the tuning dataset (lines 3-6). We also show the results when tuning on the full tuning dataset (line 7), as well as what the PRO-fixes proposed in (Nakov et al., 2012) would achieve when tuning on the full dataset (lines 1-2).

We then built a phrase table using the Moses pipeline with max-phrase-length 7 and Kneser-Ney smoothing, as well as a lexicalized reordering model (Koehn et al., 2005): *msd-bidirectional-fe*. We used a 5-gram language model trained on GigaWord v.5 with Kneser-Ney smoothing using KenLM (Heafield, 2011). On tuning and testing, we dropped the unknown words.

For tuning, we used PRO. In order to avoid instabilities when tuning on long sentences, we used a slightly modified, fixed version of PRO, as we recommended in (Nakov et al., 2013b), where we limited the difference between the positive and the negative example in a training sentence pair to be no more than ten BLEU+1 points. Moreover, in order to ensure convergence, we let PRO run for up to 25 iterations (default: 16); we further used 1000-best lists in each iteration (default: 100).

In our experiments, we performed three reruns of parameter optimization, and we report BLEU averaged over the three reruns, as suggested by Clark et al. (2011) as a way to stabilize MERT. We calculated BLEU using NIST's scoring tool v.13a, in case-sensitive mode.

5 Experiments and Evaluation

In this section, we verify experimentally whether tuning on short sentences can make PRO's length issue worse and whether tuning on longer sentences could help in that respect. We further compare the effect of tuning on long sentences (i.e., of tuning dataset customization) to using better smoothing for BLEU+1 (as we have proposed in our earlier work).

Lines 3-6 in Table 1 show the results when tuning on the longest (top50), middle (mid50), random (rand50) and shortest (low50) 50% of the tuning sentences. Comparing this to line 7 (tuning on the full MT06), we can see that tuning on the shortest sentences lowers the hypothesis-to-reference ratio (BP), while tuning on top50 improves it,[3] with the BP for mid50 and rand50 in between.

Lines 3-6 further show that better BP corresponds to better BLEU. We can also see that both BP and BLEU for top50 are better than those for the full MT06 tuning dataset. Despite top50 being tuned on less data, its BP and BLEU are comparable to those achieved by the BLEU+1 smoothing approaches shown in lines 1-2 (Nakov et al., 2012), which use the full tuning dataset.

Note that when calculating BP and BLEU, for the 4-reference MT06 dataset, we used the length of the reference sentence that is closest to the length of the hypothesis. This is the *effective reference length* from the original paper on BLEU (Papineni et al., 2002), and it is also the default in NIST scoring tool v13a, which we use.

Using the closest reference yields a very forgiving BP. Yet, few datasets have multiple references. Thus, we also experimented with a single (ref0) reference for both tuning and testing. The results are shown in Table 2. Comparing the corresponding lines of Tables 2 and 1, we see that in this case, the length problem is more severe and affects BLEU more. More importantly, note that the top50 customized tuning dataset is much more effective with a single reference translation.

[3]The ideal target value for BP is 1.

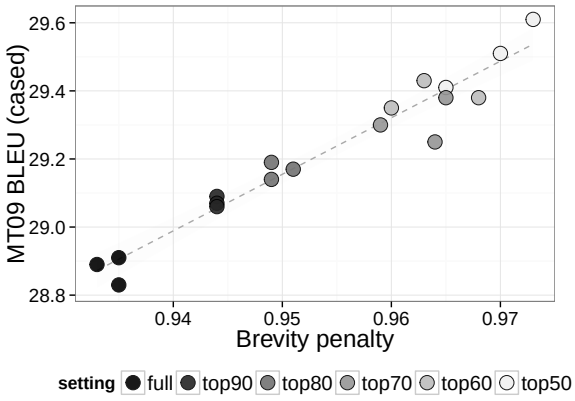

Figure 1: Single-reference PRO experiments. Correlation between cutoff, BP and BLEU score. The x-axis shows the brevity penalty (BP), and the y-axis contains the BLEU score on the testing MT09 dataset. Different colors show the different levels of cutoff. We show the results for three reruns in each setting.

6 Discussion

In this section, we perform further analysis, in order to better understand the improvements when tuning in longer sentences. We consider three aspects: (*i*) amount of training data, (*ii*) genre overlap between tuning and test datasets, and (*iii*) differences in the learned SMT parameters.

6.1 Amount of Tuning Data

In our experiments, we saw that tuning on longer sentences yields better results than tuning on shorter ones. However, one might argue that the subsets with longer sentences have access to more training data in terms of number of word tokens. In order to shed some light on this, we experimented with varying the percentage of longest sentences that we keep in decreasing order: from the full dataset (100%), we gradually removed the shortest sentences in increments of 10% until we ended up with just 50% of the data. The results are shown in Figure 1. We can see that as the cutoff increases, so does BP, which in turn yields better BLEU. This suggests that by varying the length of the tuning sentences, we can effectively control the verbosity that PRO learns. We can further conclude that it is not the amount of tuning data that matters but rather its characteristics.

6.2 Genre Overlap

MT06 is a mixture of three genres: newswire (nw), weblogs (wb) of almost equal sizes, and a much smaller size of broadcast news (bn). MT09 is also a mixture, but of two genres only: it only contains newswire and weblogs. Thus, one could ask the question of whether the observed improvements are due to better overlap between the genres of the tuning and of the testing datasests.

	bn	nw	wb	D_{KL}
MT06				
full	8%	**46**%	**46**%	3.93
low50	7%	25%	64%	7.42
mid50	9%	**46**%	**41**%	6.41
top50	8%	63%	26%	12.09
MT09				
full		**45**%	**55**%	

Table 3: Distribution of genres for the different partitions of the tuning data (MT06) and the test data (MT09). While MT06 has newswire (nw), and weblogs(wb) in equal amounts (with a lower proportion of broadcast news (bn)), MT09 has slightly higher proportion of weblog data than newswire. Based on Kullback–Leibler divergence (D_{KL}), the full partition is closest to the test data, followed by the mid50 partition.

Table 3 could help answer this question; it shows the genre distribution for the different partitions of the tuning dataset. We can see that the distribution of genres in the full MT06 dataset is better than for top50, mid50, and low50, having the smallest Kullback–Leibler divergence with the test set; the mid50 partition comes second. In contrast, top50, the best-performing partition among the ones we explored, has the most divergent genre-distribution with respect to the test dataset. From this, we can conclude that the improvements in BLEU for top50 are definitely not due to better genre/domain overlap.

6.3 Better Parameters

The last question we address in this section is the following: Where exactly is the difference in translation quality coming from? I.e., are the improvements in length only the result of decrease in the word penalty or are there other parameters that are being affected? In order to answer this question, we analyzed the optimized weights (averaged over three reruns) for tuning at different cutoffs, from 100% to 50% of the longest sentences in the MT06 development dataset.

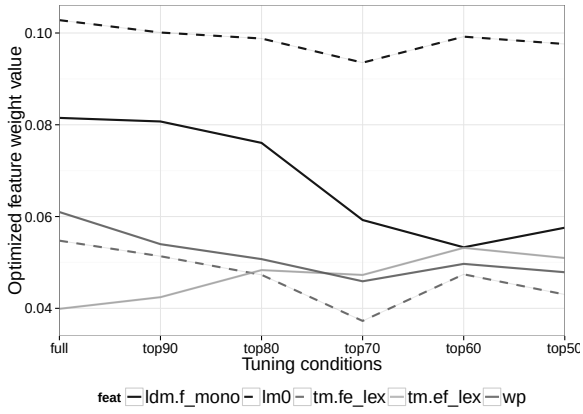

Figure 2: Optimized feature weight values for each of the different tuning settings. Only a subset out of the 14 different tuning weights that are used by the SMT model, and are thus being optimized, are shown in the figure, namely the following: monotone lexicalized reordering (ldm.f_mono), language model (lm0), reverse lexical phrase translation probabilities (tm.fe_lex), direct lexical phrase translation probabilities (tm.ef_lex), and word penalty (wp).

We selected the most important feature weights in terms of their correlation with changes in the brevity penalty (and BLEU). The results are shown in Figure 2. As expected, the average value for the word penalty weight (dark gray solid line) is reduced as we increase the average length of our tuning set. This results in lower costs for longer sentences, explaining why we have higher verbosity.

However, this is not the full picture. The monotone lexicalized reordering model (black solid line) sees significant reduction in its weight, allowing for more reordering. Furthermore, the weight for the direct lexical phrase translation (light gray line) slightly increases as we increase the length of our tuning data. This can be interpreted as increased reliance on word-to-word translations.

Thus, by changing the length of the development set, we not only affect the word penalty, but also allow for changes in other parameters, which jointly yield better translation.[4]

[4]To be more precise in this analysis, more careful study needs to be done using the *expected decoding cost*, i.e., multiplying the optimized weights by the mean feature values on a specific set. Nonetheless, there is no clear way to obtain such a mean feature vector without using a specific set of weights for decoding in the first place.

7 Conclusion and Future Work

We have explored the idea of customizing the tuning dataset for Statistical Machine Translation (SMT) that makes the hyper-parameters of the SMT system more robust with respect to some specific deficiencies of the parameter tuning algorithms. This is an under-explored research direction, which can allow better parameter tuning. In this paper, we achieved this goal by selecting a subset of the available sentence pairs, which are more suitable for specific combinations of optimizers, objective functions, and evaluation measures. In particular, we experimented with the pairwise ranking optimization (PRO) optimizer, which is known to yield too short translations. We have shown that the problem can be alleviated by tuning on a subset of the development dataset, selected based on sentence length. In particular, when selecting the longest 50% of the tuning sentences, we achieved two-fold tuning speedup and competitive scores in terms of BLEU, while having a more compact dataset. These results rival those of alternative solutions that we proposed in our previous work, which fix the tuning-time BLEU+1 smoothing instead. Our analysis shows that this is due to improved parameter tuning.

Overall, our goal was not just to show how one can improve PRO, but rather to draw the research attention to the more general idea of customizing a tuning set through subset selection, which can offer a number of opportunities for different experimental conditions, and more efficient training. We believe that this is a very promising research direction, which is worth exploring further.

In the future, we plan to experiment with other language pairs and translation directions, as well as with other optimizers such as MERT and MIRA (instead of PRO), and with other evaluation measures such as TER (Snover et al., 2006), METEOR (Lavie and Denkowski, 2009), and DiscoTK (Joty et al., 2014), including also pairwise measures (Guzmán et al., 2015a) (instead of BLEU). We further want to study the sensitivity of these optimizer and metric combinations with respect to length and other characteristics of the tuning dataset, which would allow us to design targeted dataset customization strategies for them.

Acknowledgements

We would like to thank the anonymous reviewers for their constructive comments.

References

Marzieh Bazrafshan, Tagyoung Chung, and Daniel Gildea. 2012. Tuning as linear regression. In *Proceedings of the 2012 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Montréal, Canada, NAACL-HLT '12, pages 543–547.

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*. New York, NY, USA, HLT-NAACL '06, pages 17–24.

Colin Cherry and George Foster. 2012. Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Montréal, Canada, NAACL-HLT '12, pages 427–436.

David Chiang, Kevin Knight, and Wei Wang. 2009. 11,001 new features for statistical machine translation. In *Proceedings of the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Boulder, Colorado, USA, NAACL-HLT '09, pages 218–226.

David Chiang, Yuval Marton, and Philip Resnik. 2008. Online large-margin training of syntactic and structural translation features. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, Hawaii, USA, EMNLP '08, pages 224–233.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA, ACL '11, pages 176–181.

Michael Denkowski and Alon Lavie. 2011. Meteor-tuned phrase-based SMT: CMU French-English and Haitian-English systems for WMT 2011. Technical report, CMU-LTI-11-011, Language Technologies Institute, Carnegie Mellon University.

Chris Dyer, Kevin Gimpel, Jonathan H. Clark, and Noah A. Smith. 2011. The CMU-ARK German-English translation system. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Edinburgh, Scotland, UK, WMT '11, pages 337–343.

Kevin Gimpel and Noah A. Smith. 2012. Structured ramp loss minimization for machine translation. In *Proceedings of the 2012 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Montréal, Canada, NAACL-HLT '12, pages 221–231.

Francisco Guzmán, Shafiq Joty, Lluís Màrquez, and Preslav Nakov. 2015a. Pairwise neural machine translation evaluation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*. Beijing, China, ACL-IJCNLP '15, pages 805–814.

Francisco Guzmán, Preslav Nakov, Ahmed Thabet, and Stephan Vogel. 2012. QCRI at WMT12: Experiments in Spanish-English and German-English machine translation of news text. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*. Montréal, Canada, IWSLT '12, pages 298–303.

Francisco Guzmán, Preslav Nakov, and Stephan Vogel. 2015b. Analyzing optimization for statistical machine translation: MERT learns verbosity, PRO learns length. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*. Beijing, China, CoNLL '15, pages 62–72.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Edinburgh, Scotland, UK, WMT '11, pages 187–197.

Mark Hopkins and Jonathan May. 2011. Tuning as ranking. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, Scotland, UK, EMNLP '11, pages 1352–1362.

Shafiq Joty, Francisco Guzmán, Lluís Màrquez, and Preslav Nakov. 2014. DiscoTK: Using discourse structure for machine translation evaluation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA, WMT '14, pages 402–408.

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *Proceedings of the International Workshop on Spoken Language Translation*. Pittsburgh, Pennsylvania, USA, IWSLT '05, pages 68–75.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*. Prague, Czech Republic, ACL '07, pages 177–180.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. 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. Edmonton, Canada, HLT-NAACL '03, pages 48–54.

Alon Lavie and Michael J. Denkowski. 2009. The METEOR metric for automatic evaluation of machine translation. *Machine Translation* 23:105–115.

Mu Li, Yinggong Zhao, Dongdong Zhang, and Ming Zhou. 2010. Adaptive development data selection for log-linear model in statistical machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*. Beijing, China, COLING '10, pages 662–670.

Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie Dorr. 2007. Using paraphrases for parameter tuning in statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic, pages 120–127.

Nitin Madnani, Philip Resnik, Bonnie J. Dorr, and Richard Schwartz. 2008. Are multiple reference translations necessary? Investigating the value of paraphrased reference translations in parameter optimization. In *Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas*. Waikiki, Hawaii, AMTA '08'.

David McAllester and Joseph Keshet. 2011. Generalization bounds and consistency for latent structural probit and ramp loss. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*. Granada, Spain, NIPS '11, pages 2205–2212.

Preslav Nakov. 2008. Improved statistical machine translation using monolingual paraphrases. In *Proceedings of the 18th European Conference on Artificial Intelligence*. Amsterdam, The Netherlands, ECAI '08, pages 338–342.

Preslav Nakov, Fahad Al Obaidli, Francisco Guzmán, and Stephan Vogel. 2013a. Parameter optimization for statistical machine translation: It pays to learn from hard examples. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*. Hissar, Bulgaria, RANLP '13, pages 504–510.

Preslav Nakov, Francisco Guzmán, and Stephan Vogel. 2012. Optimizing for sentence-level BLEU+1 yields short translations. In *Proceedings of the 24th International Conference on Computational Linguistics*. Mumbai, India, COLING '12, pages 1979–1994.

Preslav Nakov, Francisco Guzmán, and Stephan Vogel. 2013b. A tale about PRO and monsters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, ACL '13, pages 12–17.

Preslav Nakov and Hwee Tou Ng. 2009. Improved statistical machine translation for resource-poor languages using related resource-rich languages. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. Singapore, EMNLP '09, pages 1358–1367.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA, ACL '02, pages 311–318.

Ryan Roth, Owen Rambow, Nizar Habash, Mona Diab, and Cynthia Rudin. 2008. Arabic morphological tagging, diacritization, and lemmatization using lexeme models and feature ranking. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Columbus, OH, USA, ACL '08, pages 117–120.

Patrick Simianer, Stefan Riezler, and Chris Dyer. 2012. Joint feature selection in distributed stochastic learning for large-scale discriminative training in SMT. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*. Jeju Island, Korea, ACL '12, pages 11–21.

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 the Conference of the Association for Machine Translation in the Americas*. Cambridge, Massachusetts, USA, AMTA '06, pages 223–231.

Aleš Tamchyna, Petra Galuščáková, Amir Kamran, Miloš Stanojević, and Ondej Bojar. 2012. Selecting data for English-to-Czech machine translation. In *Proceedings of the Workshop on Statistical Machine Translation*. Montréal, Canada, WMT '12, pages 374–381.

Taro Watanabe, Jun Suzuki, Hajime Tsukada, and Hideki Isozaki. 2007. Online large-margin training for statistical machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Prague, Czech Republic, EMNLP-CoNLL '07, pages 764–773.

Zhongguang Zheng, Zhongjun He, Yao Meng, and Hao Yu. 2010. Domain adaptation for statistical machine translation in development corpus selection. In *Proceedings of the 4th International Universal Communication Symposium*. Beijing, China, IUCS '10, pages 2–7.