Integrating Linguistic Information in Machine Translation Evaluation

Jason Adams
jmadams@cs.cmu.edu
Advanced Machine Translation Seminar White Paper

May 5, 2008

1 Introduction

The automatic evaluation of machine translation (MT) has been a very important factor driving the success of statistical machine translation for most of this decade. Prior to automatic metrics, researchers were forced to rely more heavily on human evaluations, which are costly and time-consuming. Automatic metrics allow systems to analyze and reduce errors while they train. Fully automatic machine translation evaluation poses an interesting problem, however. In order to be able to judge whether a translation from a source to target language is a good one, we must first solve the problem of machine translation. For that reason, human reference translations are created for the purpose of evaluation and the target output is compared against these reference translations. Some metrics also look at the source sentences for length ratios or other statistics.

The success of automatic MT evaluation relies heavily upon accurate reference translations. There are at least two assumptions in dealing with reference translations: that the human translation will be adequate in the transmission of information from the target language to the source language and that the human translation will be fluent in the target language. It is possible for there to be a many-to-many mapping from sentences with the same meaning in the source language to sentences with the same meaning in the target language. Also, it is still a matter of debate in the linguistic community whether exact meaning-to-meaning mapping is even possible between human languages (cf. the Sapir-Whorf hypothesis (Kay and Kempton, 1984)). Another complication is deciding how best to evaluate the quality of the MT evaluation metrics themselves.

These factors make automatic MT evaluation a difficult task. Reference translations simplify it, but how best to exploit them is still an open question. This paper will begin by briefly mentioning current popular metrics for MT evaluation and how MT metrics are evaluated in Section 2. In Section 3, we will describe some recent attempts of incorporating linguistic information in evaluation metrics and in Section 4, we will see explore how machine learning has been combined with this information. Finally, we will conclude in Section 5 with some final discussion and ideas for future research in this field.

2 Background

Evaluating MT evaluation is a growing research area in itself. One common evaluation strategy has been to collect human judgments of MT quality for measures of adequacy and fluency. A bilingual judge examines the source sentence and reference translations and assigns a rating (usually in the scale of 1 to 5) to that sentence for both of these measures. There is a great deal of intercoder disagreement (Koehn and Monz, 2006), so these human assessments are usually normalized according to the method described in (Blatz et al., 2003). The MT evaluation metric is evaluated based on correlation (using Pearson correlation or Spearman's rank correlation) with the judges' normalized assessments.

2.1 Binary System Comparisons

Intercoder disagreement between judges using rating scales indicates a need for a better method of scoring hypothesis translations. One alternative for human evaluation of multiple machine translation systems is binary system comparison (Vilar et al., 2007). The central insight of binary system comparison is that a human judge is better able to tell which of two translations is of higher quality than he is of assigning a rating to each. The process is simple: a judge is presented with two translations of the same source sentence and is instructed to mark which sentence is better or if they are equal. It is up to each judge to decide what it means to be the better translation, and they are instructed not to distinguish between adequacy and fluency. Judges are presented with sentences from multiple systems chosen at random, to alleviate the effect of bias in the evaluators.

Once enough binary comparisons for multiple systems have been elicited, a ranking of all the systems is possible. Vilar et al. (2007) define the comparison score between two systems X and Y, with translations $e_{i,X}$ and $e_{i,Y}$

as

$$r_{i,X,Y} = \begin{cases} +1 & e_{i,X} \text{ is better than } e_{i,Y} \\ 0 & e_{i,X} \text{ is equal to } e_{i,Y} \\ -1 & e_{i,X} \text{ is worse than } e_{i,Y} \end{cases} . \quad (1)$$

Therefore, the binary comparison score for two systems for m sentences is

$$R_{X,Y} = \frac{1}{m} \sum_{i=1}^{m} r_{i,X,Y}. \quad (2)$$

These scores also allow easy computation of standard error and significance tests. In an evaluation, Vilar et al. (2007) report that BLEU (see Section 2.2) was able to correctly rank the systems.

One major drawback of the binary system comparison approach is that it requires $O(\log n!)$ evaluations (n is the number of systems), whereas only $O(n)$ are needed for the earlier method. However, the actual task of evaluation should be easier for judges and this method does not require judges to learn a special rating scale (if one were specified). The bias introduced by the rating scale has been removed, but there is still is bias introduced by human preferences, which has not changed. The results of the rating scale method cannot be compared to future evaluations. The binary system comparison approach can be applied to later evaluations, but it requires some additional human effort. If the goal is finding the new ordering of all systems, it will require comparing the new one to several of the previous systems, which would be time consuming. However, if the goal is only to determine if the new system is significantly better than a previous system, this approach makes that fairly easy to compute.

2.2 Current Metrics

Perhaps the most widely used automatic MT evaluation metric is BLEU (Papineni et al., 2002). BLEU is concerned primarily with n-gram precision between the hypothesis and reference translations. It does not look at recall, due to the fact that it is not clear how recall should be computed across multiple references. BLEU has been often criticized in recent years as its weaknesses are being uncovered. One weakness is that in ignoring recall, it ignores a factor which is more closely correlated with human judgments (Lavie et al., 2004). Hypotheses that are scored equally by judges can vary greatly when scored by BLEU (Callison-Burch et al., 2006). BLEU also lacks flexibility when tuning to a particular task and while correlation is better at the corpus level, it correlates poorly at the sentence level (Kulesza and Shieber, 2004; Och et al., 2003).

METEOR is another metric that seeks to address many of the concerns in BLEU and increase correlation with human judgments at the corpus and sentence level (Banerjee and Lavie, 2005; Lavie and Agarwal, 2007). METEOR addresses the issue of recall by using the maximum recall between the hypothesis and each reference translation. Like BLEU, it is an n-gram measure, but it also brings in additional semantic information if the optional WordNet synonymy module is used. This has been shown independently to increase correlation with human judgments (for example, (Owczarzak et al., 2007)). Using WordNet to produce synonyms for n-gram matching is one way of bringing linguistic information into MT evaluation. In the next section, we will look at methods that take it a step further.

3 Incorporation of Linguistic Information

The past few years have seen the appearance of several MT evaluation techniques that use linguistic information. Most often, the linguistic information is syntactic, and compares automatically obtained parses of the hypothesis and reference translations. One of the first experiments in this area was by Liu and Gildea (2005). It is possible for BLEU to assign high scores to disfluent sentences that would have been caught if the syntax of the sentences were known. Some of the issues that linguistically motivated approaches face are noise. Parses for sentences are obtained via automatic methods. Parsing accuracy is typically in the 90% accuracy range, but that still means there may be several errors in each sentence. The sensitivity to noise of a metric is an important consideration. Most automatic parsers are trained and evaluated on data that has been constructed by hand. How they degrade in the face of poorly formed sentences can also affect performance. N-gram metrics like BLEU are usually very fast, while parsing is much slower. Minimum error rate training for machine translation is only possible (in a reasonable time) with a very fast evaluation measure. If syntactic methods are to benefit MT during training, they must also be efficient. This particular concern has not really been addressed in the literature yet and is an open area for research.

3.1 Subtree Features

The metrics described by Liu and Gildea (2005) all examine the similarity between subtrees of the hypothesis and reference translations. Matching whole trees is improbable (unless the sentences were identical, in which case

the extra effort is unnecessary). Subtree similarity uses syntactic information and allows degrees of similarity to be expressed in a more fine-grained way. This method has intuitive merit in that it also rewards synonymy in a way that would be missed by an n-gram method. Assuming the higher levels of the trees stayed the same, synonyms and synonymous phrases can exchange leaf node positions. At the same time, this may admit additional errors, since it is also possible for incorrect words to fill those positions. The hypothesis is that the latter case will be less likely in general than the former. It is important to note that syntactic features are more likely to be helpful in identifying fluency rather than adequacy. However, a fluent sentence that scores well with reference translations should also have a high level of adequacy.

The first metric they propose is the subtree metric (STM). A subtree of depth n in this metric is a parent node and all of its children and their children down to a depth of n. Clipped precision is computed on all subtrees of a given depth and normalized by the maximum depth of the tree to produce the STM score:

$$STM = \frac{1}{D}\sum_{i=1}^{D}\frac{\sum_{t\in\text{subtrees}_n}(\text{hyp})\text{count}_{\text{clip}}(t)}{\sum_{t\in\text{subtrees}_n}(\text{hyp})\text{count}_{(t)}}. \quad (3)$$

The count is the number of times the subtree t appears in the hypothesis tree, while the clipped count is the number of times the subtree appears in the reference translations capped by the maximum number of times it appears in any single reference. This prevents assigning high scores to trees that produce too many common subtrees. A similar problem occurs when calculating unigram precision. The sentence *The the the the the* would have a high precision as long as *the* appeared in the reference translation.

Rather than constructing subtrees that contain all of a parent's children, they also considered all fractional parts of subtrees. To make this tractable, they used convolution kernels as described by Collins and Duffy (2002). Kernel methods operate on vectors, which in this case are vectors of counts of all occurrences of every subtree in the hypothesis and reference translations. The inner product of the hypothesis and a reference vector is a count of the number of subtrees in common. This value is normalized by the magnitude of each vector to produce the cosine of the two vectors, which is a value between between 0 and 1. The resulting Tree Kernel Metric (TKM) is the maximum cosine between the hypothesis and any reference.

The previous two metrics used phrase structure parses similar to that used by the Penn Treebank. For the third metric, they turn to dependency

parses of the sentences. In a dependency parse, branches of the tree are no longer labeled by part of speech, but simply link children to parents according to syntactic dependency. This introduces the notion of headedness. That is, phrases are headed by a root word. For a noun phrase such as *the silly doctor*, the head word would be *doctor* and both *the* and *silly* would be its children. Any chain of dependencies, from child to parent, is considered a headword chain. The Headword Chain Metric (HWCM) is computed in exactly the same fashion as the STM, except that any headword chain is permissible. Recall that in the STM, a parent and *all* of its children constituted a subtree. Liu and Gildea (2005) also evaluated dependency versions of STM and TKM (created with dependency parses rather than phrase structure parses).

To evaluate these metrics, they used data with human assessments of adequacy and fluency on a scale of 1 to 5. The data came from the Johns Hopkins 2003 Summer Workshop and the ACL05 MT Workshop. At both the corpus level and sentence level, HWCM correlated better with human judgments than did BLEU. At the corpus level, the dependency versions of STM and TKM both generally outperformed BLEU. Only the kernel based metrics did worse at the sentence level. These results were valuable in that they showed that even noisy linguistic information is helpful in judging the quality of MT output.

3.2 Grammatical Relations

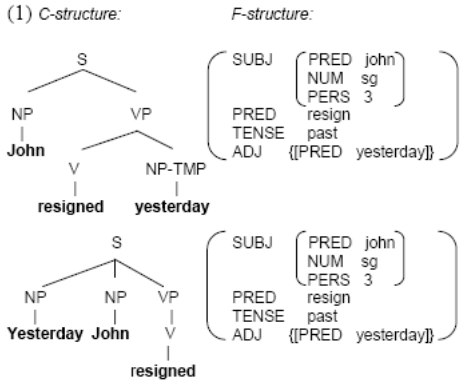


Figure 1: An example LFG parse (Owczarzak et al., 2007).

Owczarzak et al. (2007) reported an interesting extension of the metrics proposed by Liu and Gildea (2005). Rather than using a standard

phrase structure parser, they used a Lexical Functional Grammar (LFG) parser developed by Cahill et al. (2004). LFG is a grammar formalism which constructs at least two structures for each sentence: a c-structure and an f-structure. The c-structure is similar to a parse tree, but the f-structure describes the grammatical relations present in the sentence. A grammatical relation is a mapping of some grammatical role to the words acting as such in the sentence. An example sentence from Owczarzak et al. (2007) is given in Figure 1. Whereas Liu and Gildea (2005) used only the constituent information (equivalent to the LFG c-structure), Owczarzak et al. (2007) looked also at the grammatical relations present in the hypothesis and reference translations.

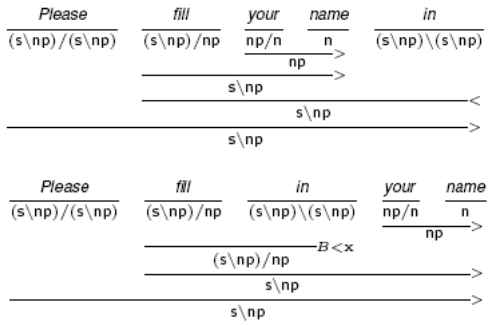
The grammatical relations they examined can be grouped into *predicate-only* and *non-predicate* dependencies. Predicate-only dependencies are predicate-value pairs (e.g. *subj*(resign, john)). Non-predicate dependencies are typically features that indicate the degree or the form of words in the sentence. Rather than describing the relationship between words, they describe the attributes of words and the sentence as a whole. Example non-predicate features include adjectival degree, complementizer forms (if, whether), passive, infinitival clause, etc. Whereas subtree features looked at the similarities between subtrees in the parses of the reference and hypothesis translations, the labeled dependency approach measures the similarity in the f-structures of the reference and hypothesis translations. This is done by computing the harmonic mean (f-score) between precision and recall for each hypothesis, reference translation pair.

In LFG, it is possible for a single f-structure to map to many surface forms of a sentence. Intuitively, this seems like an ideal characteristic for an MT metric to have since there can be wide variation in target-side sentences while still being a good translation of the source sentence. One issue that arises whenever statistical parsing is involved is parser noise. To address this issue, Owczarzak et al. (2007) use an interesting approach. When adjuncts are reordered in a sentence, it is usually the case that the f-structure stays the same. By performing meaning-preserving transformations on a set of sentences, it is possible to compare the resulting f-structures in order to measure the noise introduced by the parser. This comparison poses easily as a machine translation problem: the source sentence is the original sentence and the translation is the adjunct-reordered form of the sentence. This reformulation allows the use of other MT evaluation metrics to measure the parser noise. By adding n-best parses, the scores according to the labeled dependencies steadily improve.

The result of adding labeled dependencies appears to be beneficial. Owczarzak

et al. (2007) report results that outperform all other examined metrics (including METEOR with WordNet) on Pearson correlation with unnormalized human judgments of fluency. METEOR correlates the best with adequacy and with the average of adequacy and fluency, though the labeled dependency approach is competitive on average, when using the 50-best parses, and it significantly outperforms BLEU. The labeled dependencies used are still fairly simplistic and they describe future work with more complex features built from the parser output.

3.3 Syntactic Word-Word Dependencies



(det name$_3$ your$_2$) (det name$_4$ your$_3$)
(dobj fill$_1$ name$_3$) (dobj fill$_1$ name$_4$)
(ncmod _ fill$_1$ in$_4$) (ncmod _ fill$_1$ in$_2$)
(xcomp _ please$_0$ fill$_1$) (xcomp _ please$_0$ fill$_1$)

Figure 2: Example parses of two sentences using CCG (Mehay and Brew, 2007)

.

BLEUATRE (BLEU's Associate with Tectogrammatical RElations) is a different twist on incorporating syntax into MT evaluation (Mehay and Brew, 2007). Rather than parsing both the noisy hypothesis and reference translations, BLEUATRE parses only the reference translations. By virtue of the fact that these are human generated, they are expected to be well-formed, side-stepping the problem parsers face with noisy input. To extract dependencies, they use a Combinatory Categorial Grammar (CCG) parser developed by Clark and Curran (2004). A word-word dependency is a tuple consisting of the head word, the dependent (child) word, and the part of

speech of the child. An example pair of sentences along with their word-word dependencies is given in Figure 2.

Once the word-word dependencies have been extracted for the reference translations, two lists are constructed: the words that must appear to the left of the heads and the words that must appear to the right. This allows a partial linear order to be established in the dependency parse. To evaluate the performance of an MT system, the candidate sentences are evaluated on how often it gets the head-dependent ordering correct. This requires no parsing, only simple n-gram comparisons like BLEU. Also like BLEU, BLEUATRE uses a brevity penalty to protect against overgeneration in sentences that are too long.

While BLEUATRE does not achieve the best results on correlation with human judgment, it does at least show that it is possible to achieve competitive results without parsing the hypothesis translations. This offers hope that a syntactic method could be incorporated in minimum error rate training for MT systems. Also, as Mehay and Brew (2007) point out, the BLEUATRE system does not include many of the enhancements (such as WordNet synonymy) that other high-scoring metrics do. The BLEUATRE system could easily be combined with other systems or linguistically informed approaches, perhaps boosting the correlation of both.

4 Combination with Machine Learning

A different way of looking at the MT evaluation task is as a human-likeness problem. Rather than attempting to judge the similarity between the hypothesis and reference translation, we could build a classifier that uses those similarity measures as features to determine whether a sentence was produced by a human or a machine (Corston-Oliver et al., 2001; Kulesza and Shieber, 2004). The features looked at in their experiment were similar to those used by BLEU and other n-gram metrics. In addition to n-gram precision, they looked at the ratio of hypothesis to reference sentence length, word error rate, and position-independent word error rate. Reference translations were treated as positive training examples and machine translations were treated as negative. They trained a support vector machine (SVM) to classify test sentences according to their human likeness and found that it performed very well, correlating much better with human judgments at the sentence level than did BLEU.

After the work by Liu and Gildea (2005), an interesting recent extension has been to use subtree features (specifically HWCM) as additional features

in the SVM model (Albrecht and Hwa, 2007). The paper also looks at whether classification and the way the classification question was framed is the right approach. They ask three important questions:

- Is human-likeness the correct machine learning approach for MT evaluation?
- How do the model features affect the result?
- Do learning approaches generalize?

To answer the first question, they compare the human-likeness classifier of Kulesza and Shieber (2004) with a regression-based model. In regression, rather than trying to classify a sentence as coming from a human or machine, it attempts to learn the continuous function matching how humans assess quality. In their experiments, Albrecht and Hwa (2007) show that achieving a high level of accuracy on human-likeness classification does not necessarily mean better correlation with human judgment. Regression correlated better with human judgments and was able to do so given available training data.

Two sets of model features were constructed in order to determine the effect they would have on the result of the regression model. The small feature set were the features from Kulesza and Shieber (2004). The features from that work were then applied to headword chains derived in the same fashion as Liu and Gildea (2005). The features from the small feature set were computed with a large corpus of English acting as a reference translation and repeated for the syntactic features over the larger dataset. The results showed that having more features helped regression as the number of training examples increased, whereas it had a large, negative impact on classification at all levels of training data. Interestingly, the classification accuracy of the classifier increased with more features and training examples, but correlation with human judgments did not.

To determine how well regression generalizes to unseen data, they performed a series of experiments across years on NIST evaluation data. They compare the cross-year generalization performance of regression, classification, BLEU, METEOR, and HWCM. Regression performed better than any other metric, demonstrating that it does generalize well. In another experiment, the top and bottom metrics were removed from the training data and the results for each run were compared to the results when all of the data was present in training. The results from those experiments indicated that the presence of good examples in the training data is very important. Good examples are more constrained and so provide more information to the learner than bad examples, which can vary widely in form.

5 Conclusion

Results from recent work incorporating linguistic information into MT evaluation metrics have shown that linguistic information is helpful in improving correlation with human judgments. Finding the best way to incorporate this information is not always straightforward. Also, n-gram similarity metrics are usually faster and achieve results that are good enough (in the case of BLEU) or very good (in the case of METEOR) without having to expend the extra effort of using linguistic information. Also, automatic methods of obtaining this information is often very noisy and the data used as input (MT system output) is not the same as the data that these parsers were trained on. To our knowledge, no oracle studies have been done on determining just how well these metrics would perform with perfect linguistic knowledge. To do so would be extraordinarily time consuming and require the effort of several trained experts.

Another drawback of including linguistic information is the computational cost. In general, it is prohibitively high for use in minimum error rate training for an MT system. The work by Mehay and Brew (2007) with the BLEUATRE system offers some possibilities in this area, however. It is still an open question whether the use of a metric with better correlation with human judgments would be beneficial for that purpose.

Machine learning has only begun to be used in MT evaluation. We expect that in upcoming years, we will see more and more systems which attempt to build good machine learning models for this task. Linguistic information has been applied in this area already to good effect, so we expect that trend to continue. The obvious next steps would be to use labeled dependencies in a system such as that used by Albrecht and Hwa (2007). The most common machine learning algorithm has been support vector machines, but it's possible there are better learners. Also, regression proved to be better than human-likeness classification, so it is possible there are still better ways of framing the evaluation question for machine learning. Perhaps by using binary system comparisons, a ranking SVM (see (Joachims, 2002) could be trained with linguistically motivated features. In any case, there are many interesting years ahead for research in MT evaluation.

References

Albrecht, J. and Hwa, R. (2007). A Re-examination of Machine Learning Approaches for Sentence-Level MT Evaluation. In *Proceedings of the 45th*

Annual Meeting of the Association for Computational Linguistics.

Banerjee, S. and Lavie, A. (2005). METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL 2005 Workshop on Intrinsic and Extrinsic Evaluation measures for MT and/or Summarization.*

Blatz, J., Fitzgerald, E., Foster, G., Gandrabur, S., Goutte, C., Kulesza, A., Sanchis, A., and Ueffing, N. (2003). Confidence Estimation for Machine Translation. Technical report, JHU / CLSP Summer Workshop.

Cahill, A., Burke, M., O'Donovan, R., van Genabith, J., and Way, A. (2004). Long-distance dependency resolution in automatically acquired wide-coverage pcfg-based lfg approximations. *Proceedings of ACL*, 4:320–327.

Callison-Burch, C., Osborne, M., and Koehn, P. (2006). Re-evaluating the Role of BLEU in Machine Translation Research. In *Proceedings of EACL-2006.*

Clark, S. and Curran, J. R. (2004). Parsing the wsj using ccg and log-linear models. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 103, Morristown, NJ, USA. Association for Computational Linguistics.

Collins, M. and Duffy, N. (2002). Convolution kernels for natural language. *Advances in Neural Information Processing Systems*, 14:625–632.

Corston-Oliver, S., Gamon, M., and Brockett, C. (2001). A machine learning approach to the automatic evaluation of machine translation. *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 148–155.

Joachims, T. (2002). Optimizing search engines using clickthrough data. *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142.

Kay, P. and Kempton, W. (1984). What is the sapir-whorf hypothesis? *American Anthropologist*, 86:65–79.

Koehn, P. and Monz, C. (2006). Manual and automatic evaluation of machine translation between european languages. *Proceedings of the Workshop on Statistical Machine Translation*, pages 102–121.

Kulesza, A. and Shieber, S. (2004). A learning approach to improving sentence-level MT evaluation. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation.*

Lavie, A. and Agarwal, A. (2007). Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. *Proceedings of Workshop on Statistical Machine Translation (WMT) at the 45th Annual Meeting of the Association of Computational Linguistics (ACL-07).*

Lavie, A., Sagae, K., and Jayaraman, S. (2004). The Significance of Recall in Automatic Metrics for MT Evaluation. In *Proceedings of AMTA-2004.*

Liu, D. and Gildea, D. (2005). Syntactic Features for Evaluation of Machine Translation. In *Proceedings of the ACL 2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization.*

Mehay, D. and Brew, C. (2007). Bleuatre: Flattening syntactic dependencies for mt evaluation. *Proceedings of the 11th Conference on Theoretical and Methodological Issues in Machine Translation (TMI).*

Och, F. J., Gildea, D., Khudanpur, S., Sarkar, A., Yamada, K., Fraser, A., Kumar, S., Shen, L., Smith, D., Eng, K., Jain, V., Jin, Z., and Radev, D. (2003). Syntax for Statistical Machine Translation. Technical report, Johns Hopkins Workshop on Speech and Language Engineering.

Owczarzak, K., van Genabith, J., and Way, A. (2007). Labelled dependencies in machine translation evaluation. *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 104–111.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. In *ACL '02: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Vilar, D., Leusch, G., Ney, H., and Banchs, R. E. (2007). Human Evaluation of Machine Translation Through Binary System Comparisons. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 96–103, Prague, Czech Republic. Association for Computational Linguistics.