Syntactic Features for Evaluation of Machine Translation

Ding Liu and **Daniel Gildea**
Department of Computer Science
University of Rochester
Rochester, NY 14627

Abstract

Automatic evaluation of machine translation, based on computing n-gram similarity between system output and human reference translations, has revolutionized the development of MT systems. We explore the use of syntactic information, including constituent labels and head-modifier dependencies, in computing similarity between output and reference. Our results show that adding syntactic information to the evaluation metric improves both sentence-level and corpus-level correlation with human judgments.

1 Introduction

Evaluation has long been a stumbling block in the development of machine translation systems, due to the simple fact that there are many correct translations for a given sentence. Human evaluation of system output is costly in both time and money, leading to the rise of automatic evaluation metrics in recent years. The most commonly used automatic evaluation metrics, BLEU (Papineni et al., 2002) and NIST (Doddington, 2002), are based on the assumption that "The closer a machine translation is to a professional human translation, the better it is" (Papineni et al., 2002). For every hypothesis, BLEU computes the fraction of n-grams which also appear in the reference sentences, as well as a brevity penalty. NIST uses a similar strategy to BLEU but further considers that n-grams with different frequency should be treated differently in the evaluation. It introduces the notion of information weights, which indicate that rarely occurring n-grams count more than those frequently occurring ones in the evaluation (Doddington, 2002). BLEU and NIST have been shown to correlate closely with human judgments in ranking MT systems with different qualities (Papineni et al., 2002; Doddington, 2002).

In the 2003 Johns Hopkins Workshop on Speech and Language Engineering, experiments on MT evaluation showed that BLEU and NIST do not correlate well with human judgments at the sentence level, even when they correlate well over large test sets (Blatz et al., 2003). Kulesza and Shieber (2004) use a machine learning approach to improve the correlation at the sentence level. Their method, based on the assumption that higher classification accuracy in discriminating human- from machine-generated translations will yield closer correlation with human judgments, uses support vector machine (SVM) based learning to weight multiple metrics such as BLEU, NIST, and WER (minimal word error rate). The SVM is trained for differentiating the MT hypothesis and the professional human translations, and then the distance from the hypothesis's metric vector to the hyper-plane of the trained SVM is taken as the final score for the hypothesis.

While the machine learning approach improves correlation with human judgments, all the metrics discussed are based on the same type of information: n-gram subsequences of the hypothesis translations. This type of feature cannot capture the grammaticality of the sentence, in part because they do not take into account sentence-level information. For example, a sentence can achieve an excellent BLEU score without containing a verb. As MT systems improve, the shortcomings of n-gram based evaluation are becoming more apparent. State-of-the-art MT output

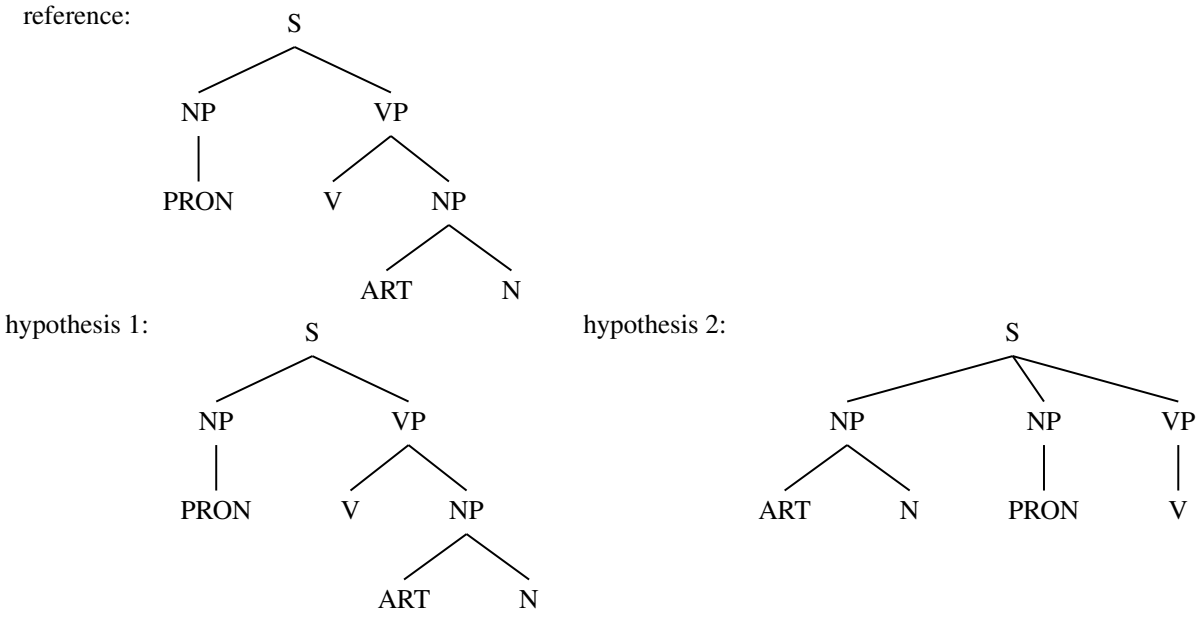


Figure 1: Syntax Trees of the Examples

often contains roughly the correct words and concepts, but does not form a coherent sentence. Often the intended meaning can be inferred; often it cannot. Evidence that we are reaching the limits of n-gram based evaluation was provided by Charniak et al. (2003), who found that a syntax-based language model improved the fluency and semantic accuracy of their system, but lowered their BLEU score.

With the progress of MT research in recent years, we are not satisfied with the getting correct words in the translations; we also expect them to be well-formed and more readable. This presents new challenges to MT evaluation. As discussed above, the existing word-based metrics can not give a clear evaluation for the hypothesis' fluency. For example, in the BLEU metric, the overlapping fractions of n-grams with more than one word are considered as a kind of metric for the fluency of the hypothesis. Consider the following simple example:

Reference: I had a dog.
Hypothesis 1: I have the dog.
Hypothesis 2: A dog I had.

If we use BLEU to evaluate the two sentences, hypothesis 2 has two bigrams *a dog* and *I had* which are also found in the reference, and hypothesis 1 has no bigrams in common with the reference. Thus hypothesis 2 will get a higher score than hypothesis 1. The result is obviously incorrect. However, if we evaluate their fluency based on the syntactic similarity with the reference, we will get our desired results. Figure 1 shows syntactic trees for the example sentences, from which we can see that hypothesis 1 has exactly the same syntactic structure with the reference, while hypothesis 2 has a very different one. Thus the evaluation of fluency can be transformed as computing the syntactic similarity of the hypothesis and the references.

This paper develops a number of syntactically motivated evaluation metrics computed by automatically parsing both reference and hypothesis sentences. Our experiments measure how well these metrics correlate with human judgments, both for individual sentences and over a large test set translated by MT systems of varying quality.

2 Evaluating Machine Translation with Syntactic Features

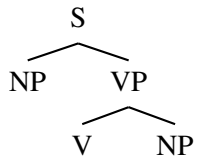
In order to give a clear and direct evaluation for the fluency of a sentence, syntax trees are used to generate metrics based on the similarity of the MT hypothesis's tree and those of the references. We can't expect that the whole syntax tree of the hypothesis can always be found in the references, thus our approach is to be based on the fractions of the subtrees

which also appear in the reference syntax trees. This idea is intuitively derived from BLEU, but with the consideration of the sparse subtrees which lead to zero fractions, we average the fractions in the arithmetic mean, instead of the geometric mean used in BLEU. Then for each hypothesis, the fractions of subtrees with different depths are calculated and their arithmetic mean is computed as the syntax tree based metric, which we denote as "subtree metric" STM:

$$\mathrm{STM} = \frac{1}{D}\sum_{n=1}^{D}\frac{\sum_{t\in \mathrm{subtrees}_n(hyp)}\mathrm{count}_{\mathrm{clip}}(t)}{\sum_{t\in \mathrm{subtrees}_n(hyp)}\mathrm{count}(t)}$$

where D is the maximum depth of subtrees considered, $\mathrm{count}(t)$ denotes the number of times subtree t appears in the candidate's syntax tree, and $\mathrm{count}_{\mathrm{clip}}(t)$ denotes the clipped number of times t appears in the references' syntax trees. Clipped here means that, for a given subtree, the count computed from the hypothesis syntax tree can not exceed the maximum number of times the subtree occurs in any single reference's syntax tree. A simple example with one hypothesis and one reference is shown in Figure 2. Setting the maximum depth to 3, we go through the hypothesis syntax tree and compute the fraction of subtrees with different depths. For the 1-depth subtrees, we get *S*, *NP*, *VP*, *PRON*, *V*, *NP* which also appear in the reference syntax tree. Since *PRON* only occurs once in the reference, its clipped count should be 1 rather than 2. Then we get 6 out of 7 for the 1-depth subtrees. For the 2-depth subtrees, we get *S→NP VP*, *NP→PRON*, and *VP→V NP* which also appear in the reference syntax tree. For the same reason, the subtree *NP→PRON* can only be counted once. Then we get 3 out of 4 for the 2-depth subtree. Similarly, the fraction of 3-depth subtrees is 1 out of 2. Therefore, the final score of STM is (6/7+3/4+1/2)/3=0.702.

While the subtree overlap metric defined above considers only subtrees of a fixed depth, subtrees of other configurations may be important for discriminating good hypotheses. For example, we may want to look for the subtree:



to find sentences with transitive verbs, while ignoring the internal structure of the subject noun phrase. In order to include subtrees of all configurations in our metric, we turn to convolution kernels on our trees. Using $H(x)$ to denote the vector of counts of all subtrees found in tree x, for two trees T_1 and T_2, the inner product $H(T_1)\cdot H(T_2)$ counts the number of matching pairs of subtrees of T_1 and T_2. Collins and Duffy (2001) describe a method for efficiently computing this dot product without explicitly computing the vectors H, which have dimensionality exponential in the size of the original tree. In order to derive a similarity measure ranging from zero to one, we use the cosine of the vectors H:

$$\cos(T_1,T_2) = \frac{H(T_1)\cdot H(T_2)}{|H(T_1)||H(T_2)|}$$

Using the identity

$$|H(T_1)| = \sqrt{H(T_1)\cdot H(T_1)}$$

we can compute the cosine similarity using the kernel method, without ever computing the entire of vector of counts H. Our kernel-based subtree metric TKM is then defined as the maximum of the cosine measure over the references:

$$\mathrm{TKM} = \max_{t\in ref}\cos(\mathrm{hyp},t)$$

The advantage of using the tree kernel is that it can capture the similarity of subtrees of different shapes; the weak point is that it can only use the reference trees one by one, while STM can use them simultaneously. The dot product also weights individual features differently than our other measures, which compute overlap in the same way as does BLEU. For example, if the same subtree occurs 10 times in both the hypothesis and the reference, this contributes a term of 100 to the dot product, rather than 10 in the clipped count used by BLEU and by our subtree metric STM.

2.1 Dependency-Based Metrics

Dependency trees consist of trees of head-modifier relations with a word at each node, rather than just at the leaves. Dependency trees were found to correspond better across translation pairs than constituent trees by Fox (2002), and form the basis of the machine translation systems of Alshawi et al. (2000)

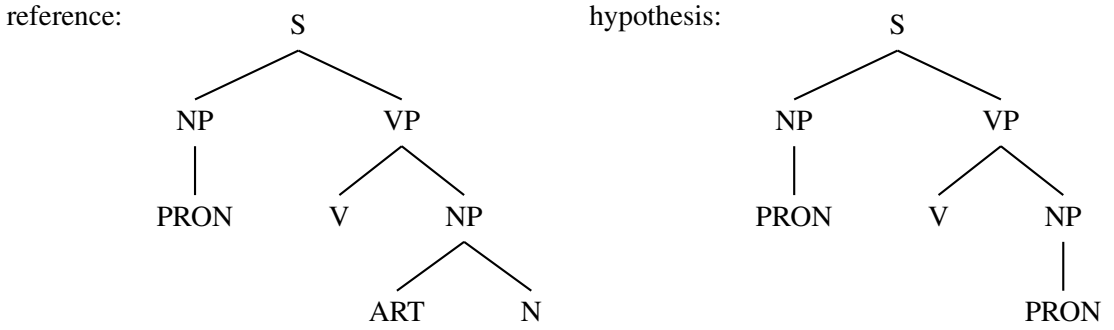
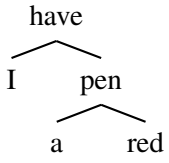


Figure 2: Examples for the Computation of STM

and Lin (2004). We derived dependency trees from the constituent trees by applying the deterministic headword extraction rules used by the parser of Collins (1999). For the example of the reference syntax tree in Figure 2, the whole tree with the root S represents a sentence; and the subtree *NP→ART N* represents a noun phrase. Then for every node in the syntax tree, we can determine its headword by its syntactic structure; from the subtree *NP→ART N*, for example, the headword selection rules chose the headword of *NP* to be word corresponding to the POS *N* in the subtree, and the other child, which corresponds to *ART*, is the modifier for the headword. The dependency tree then is a kind of structure constituted by headwords and every subtree represents the modifier information for its root headword. For example, the dependency tree of the sentence *I have a red pen* is shown as below.



The dependency tree contains both the lexical and syntactic information, which inspires us to use it for the MT evaluation.

Noticing that in a dependent tree the child nodes are the modifier of its parent, we propose a dependency-tree based metric by extracting the headwords chains from both the hypothesis and the reference dependency trees. A headword chain is a sequence of words which corresponds to a path in the dependency tree. Take the dependency tree in Figure 2 as the example, the 2-word headword chains include *have I*, *have pen*, *pen a*, and *pen red*. Before using the headword chains, we need to extract them out of the dependency trees. Figure 3 gives an algorithm which recursively extracts the headword chains in a dependency tree from short to long. Having the headword chains, the headword chain based metric is computed in a manner similar to BLEU, but using n-grams of dependency chains rather than n-grams in the linear order of the sentence. For every hypothesis, the fractions of headword chains which also appear in the reference dependency trees are averaged as the final score. Using HWCM to denote the headword chain based metric, it is computed as follows:

$$\text{HWCM} = \frac{1}{D}\sum_{n=1}^{D}\frac{\sum_{g\in \text{chain}_n(hyp)} \text{count}_{\text{clip}}(g)}{\sum_{g\in \text{chain}_n(hyp)} \text{count}(g)}$$

where D is chosen as the maximum length chain considered.

We may also wish to consider dependency relations over more than two words that are contiguous but not in a single ancestor chain in the dependency tree. For this reason, the two methods described in section 3.1 are used to compute the similarity of dependency trees between the MT hypothesis and its references, and the corresponding metrics are denoted DSTM for dependency subtree metric and DTKM for dependency tree kernel metric.

3 Experiments

Our testing data contains two parts. One part is a set of 665 English sentences generated by a Chinese-English MT system. And for each MT hypothesis, three reference translations are associated with it.

```
Input: dependency tree T, maximum length N of the headword chain
Output: headword chains from length 1 to N

for i = 1 to N
    for every node n in T
        if i == 1
            add n's word to n's 1 word headword chains;
        else
            for every direct child c of n
                for every i-1 words headword chain hc of c
                    newchain = joint(n's word, hc);
                    add newchain to the i words headword chains of n;
                endfor
            endfor
        endif
    endfor
endfor
```

Figure 3: Algorithm for Extracting the Headword Chains

The human judgments, on a scale of 1 to 5, were collected at the 2003 Johns Hopkins Speech and Language Summer Workshop, which tells the overall quality of the MT hypotheses. The translations were generated by the alignment template system of Och (2003). This testing set is called JHU testing set in this paper. The other set of testing data is from MT evaluation workshop at ACL05. Three sets of human translations (E01, E03, E04) are selected as the references, and the outputs of seven MT systems (E9 E11 E12 E14 E15 E17 E22) are used for testing the performance of our syntactic metrics. Each set of MT translations contains 929 English sentences, each of which is associated with human judgments for its fluency and adequacy. The fluency and adequacy scores both range from 1 to 5.

3.1 Sentence-level Evaluation

Our syntactic metrics are motivated by a desire to better capture grammaticality in MT evaluation, and thus we are most interested in how well they correlate with human judgments of sentences' fluency, rather than the adequacy of the translation. To do this, the syntactic metrics (computed with the Collins (1999) parser) as well as BLEU were used to evaluate hypotheses in the test set from ACL05 MT workshop, which provides both fluency and adequacy scores for each sentence, and their Pearson coefficients of correlation with the human fluency scores were computed. For BLEU and HWCM, in order to avoid assigning zero scores to individual sentences, when precision for n-grams of a particular length is zero we replace it with an epsilon value of 10^{-3}. We choose E14 and E15 as two representative MT systems in the ACL05 MT workshop data set, which have relatively high human scores and low human scores respectively. The results are shown in Table 1 and Table 2, with every metric indexed by the maximum n-gram length or subtree depth. The last row of the each table shows the tree-kernel-based measures, which have no depth parameter to adjust, but implicitly consider all depths.

Max Length/ Depth	BLEU	HWCM	STM	DSTM
1	0.126	0.130	—	—
2	0.132	0.142	0.142	0.159
3	0.117	0.157	0.147	0.150
4	0.093	0.153	0.136	0.121
kernel			0.065	0.090

Table 1: Correlation with Human Fluency Judgments for E14

The results show that in both systems our syntactic metrics all achieve a better performance in the correlation with human judgments of fluency. We also notice that with the increasing of the maximum length of n-grams, the correlation of BLEU with human judgments does not necessarily increase, but decreases in most cases. This is contrary to the argument in BLEU which says that longer n-grams better represent the sentences' fluency than the shorter

Max Length/ Depth	BLEU	HWCM	STM	DSTM
1	0.122	0.128	—	—
2	0.094	0.120	0.134	0.137
3	0.073	0.119	0.144	0.124
4	0.048	0.113	0.143	0.121
kernel			0.089	0.066

Table 2: Correlation with Human Fluency Judgments for E15

Max Length/ Depth	BLEU	HWCM	STM	DSTM
1	0.176	0.191	—	—
2	0.185	0.195	0.171	0.193
3	0.169	0.202	0.168	0.175
4	0.137	0.199	0.158	0.143
kernel			0.093	0.127

Table 3: Correlation with Human Overall Judgments for E14

Max Length/ Depth	BLEU	HWCM	STM	DSTM
1	0.146	0.152	—	—
2	0.124	0.142	0.148	0.152
3	0.095	0.144	0.151	0.139
4	0.067	0.137	0.144	0.137
kernel			0.098	0.084

Table 4: Correlation with Human Overall Judgments for E15

ones. The problem can be explained by the limitation of the reference translations. In our experiments, every hypothesis is evaluated by referring to three human translations. Since the three human translations can only cover a small set of possible translations, with the increasing of n-gram length, more and more correct n-grams might not be found in the references, so that the fraction of longer n-grams turns to be less reliable than the short ones and hurts the final scores. In the the corpus-level evaluation of a MT system, the sparse data problem will be less serious than in the sentence-level evaluation, since the overlapping n-grams of all the sentences and their references will be summed up. So in the traditional BLEU algorithm used for corpus-level evaluation, a maximum n-gram of length 4 or 5 is usually used. A similar trend can be found in syntax tree and dependency tree based metrics, but the decreasing ratios are much lower than BLEU, which indicates that the syntactic metrics are less affected by the sparse data problem. The poor performance of tree-kernel based metrics also confirms our arguments on the sparse data problem, since the kernel measures implicitly consider the overlapping ratios of the sub-trees of all shapes, and thus will be very much affected by the sparse data problem.

Though our syntactic metrics are proposed for evaluating the sentences' fluency, we are curious how well they do in the overall evaluation of sentences. Thus we also computed each metric's correlation with human overall judgments in E14, E15 and JHU testing set. The overall human score for each sentence in E14 and E15 is computed by summing up its fluency score and adequacy score. The results are shown in Table 3, Table 4, and Table 5. We can see that the syntactic metrics achieve competitive correlations in the test, among which HWCM, based on headword chains, gives better performances in evaluation of E14 and E15, and a slightly worse performance in JHU testing set than BLEU. Just as with the fluency evaluation, HWCM and other syntactic metrics present more stable performance as the n-gram's length (subtree's depth) increases.

3.2 Corpus-level Evaluation

While sentence-level evaluation is useful if we are interested in a confidence measure on MT outputs, corpus level evaluation is more useful for comparing

Max Length/ Depth	BLEU	HWCM	STM	DSTM
1	0.536	0.502	—	—
2	0.562	0.555	0.515	0.513
3	0.513	0.538	0.529	0.477
4	0.453	0.510	0.497	0.450
kernel			0.461	0.413

Table 5: Correlation with Human Overall Judgments for JHU Testing Set

Max Length/ Depth	BLEU	HWCM	STM	DSTM
1	0.629	0.723	—	—
2	0.683	0.757	0.538	0.780
3	0.724	0.774	0.597	0.780
4	0.753	0.778	0.612	0.788
5	0.781	0.780	0.618	0.778
6	0.763	0.778	0.618	0.782
kernel			0.539	0.875

Table 6: Corpus-level Correlation with Human Overall Judgments (E9 E11 E12 E14 E15 E17 E22)

MT systems and guiding their development. Does higher sentence-level correlation necessarily indicate higher correlation in corpus-level evaluation? To answer this question, we used our syntactic metrics and BLEU to evaluate all the human-scored MT systems (E9 E11 E12 E14 E15 E17 E22) in the ACL05 MT workshop test set, and computed the correlation with human overall judgments. The human judgments for an MT system are estimated by summing up each sentence's human overall score. Table 6 shows the results indexed by different n-grams and tree depths.

We can see that the corpus-level correlation and the sentence-level correlation don't always correspond. For example, the kernel dependency subtree metric achieves a very good performance in corpus-level evaluation, but it has a poor performance in sentence-level evaluation. Sentence-level correlation reflects the relative qualities of different hypotheses in a MT system, which does not indicate any information for the relative qualities of different systems. If we uniformly decrease or increase every hypothesis's automatic score in a MT system, the sentence-level correlation with human judgments will remain the same, but the corpus-level correlation will be changed. So we might possibly get inconsistent corpus-level and sentence-level correlations.

From the results, we can see that with the increase of n-grams length, the performance of BLEU and HWCM will first increase up to length 5, and then starts decreasing, where the optimal n-gram length of 5 corresponds to our usual setting for BLEU algorithm. This shows that corpus-level evaluation, compared with the sentence-level evaluation, is much less sensitive to the sparse data problem and thus leaves more space for making use of comprehensive evaluation metrics. We speculate this is why the kernel dependency subtree metric achieves the best performance among all the metrics. We can also see that HWCM and DSTM beat BLEU in most cases and exhibit more stable performance.

An example hypothesis which was assigned a high score by HWCM but a low score by BLEU is shown in Table 7. In this particular sentence, the common head-modifier relations "aboard $\leftarrow$ plane" and "plane $\leftarrow$ the" caused a high headword chain overlap, but did not appear as common n-grams counted by BLEU. The hypothesis is missing the word "fifth", but was nonetheless assigned a high score by human judges. This is probably due to its fluency, which HWCM seems to capture better than BLEU.

4 Conclusion

This paper introduces several syntax-based metrics for the evaluation of MT, which we find to be particularly useful for predicting a hypothesis's *fluency*. The syntactic metrics, except the kernel based ones, all outperform BLEU in sentence-level fluency evaluation. For the overall evaluation of sentences for fluency and adequacy, the metric based on headword chain performs better than BLEU in both sentence-level and corpus-level correlation with human judgments. The kernel based metrics, though poor in sentence-level evaluation, achieve the best results in corpus-level evaluation, where sparse data are less of a barrier.

Our syntax-based measures require the existence of a parser for the language in question, however it is worth noting that a parser is required for the target language only, as all our measures of similarity are defined across hypotheses and references in the same language.

Our results, in particular for the primarily structural STM, may be surprising in light of the fact that the parser is not designed to handle ill-formed or ungrammatical sentences such as those produced by machine translation systems. Modern statistical parsers have been tuned to discriminate good structures from bad rather than good sentences from bad.

hyp	Diplomats will be aboard the plane to return home .
ref1	Diplomats are to come back home aboard the fifth plane .
ref2	Diplomatic staff would go home in a fifth plane .
ref3	Diplomatic staff will take the fifth plane home .

Table 7: An example hypothesis in the ACL05-MTE workshop which was assigned a high score by HWCM (0.511) but a low score by BLEU (0.084). Both human judges assigned a high score (4).

Indeed, in some recent work on re-ranking machine translation hypotheses (Och et al., 2004), parser-produced structures were not found to provide helpful information, as a parser is likely to assign a good-looking structure to even a lousy input hypothesis.

However, there is an important distinction between the use of parsers in re-ranking and evaluation – in the present work we are looking for similarities between pairs of parse trees rather than at features of a single tree. This means that the syntax-based evaluation measures can succeed even when the tree structure for a poor hypothesis looks reasonable on its own, as long as it is sufficiently distinct from the structures used in the references.

We speculate that by discriminatively training weights for the individual subtrees and headword chains used by the syntax-based metrics, further improvements in evaluation accuracy are possible.

Acknowledgments We are very grateful to Alex Kulesza for assistance with the JHU data. This work was partially supported by NSF ITR IIS-09325646 and NSF ITR IIS-0428020.

References

Hiyan Alshawi, Srinivas Bangalore, and Shona Douglas. 2000. Learning dependency translation models as collections of finite state head transducers. *Computational Linguistics*, 26(1):45–60.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2003. Confidence estimation for machine translation. Technical report, Center for Language and Speech Processing, Johns Hopkins University, Baltimore. Summer Workshop Final Report.

Eugene Charniak, Kevin Knight, and Kenji Yamada. 2003. Syntax-based language models for machine translation. In *Proc. MT Summit IX*.

Michael Collins and Nigel Duffy. 2001. Convolution kernels for natural language. In *Advances in Neural Information Processing Systems*.

Michael John Collins. 1999. *Head-driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania, Philadelphia.

G. Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *In HLT 2002, Human Language Technology Conference*, San Diego, CA.

Heidi J. Fox. 2002. Phrasal cohesion and statistical machine translation. In *In Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 304–311.

Alex Kulesza and Stuart M. Shieber. 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 (TMI)*, Baltimore, MD, October.

Dekang Lin. 2004. A path-based transfer model for machine translation. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING-04)*, pages 625–630, Geneva, Switzerland.

Franz Josef Och, Daniel Gildea, Sanjeev Khudanpur, Anoop Sarkar, Kenji Yamada, Alex Fraser, Shankar Kumar, Libin Shen, David Smith, Katherine Eng, Viren Jain, Zhen Jin, and Dragomir Radev. 2004. A smorgasbord of features for statistical machine translation. In *Proceedings of the 2004 Meeting of the North American chapter of the Association for Computational Linguistics (NAACL-04)*, Boston.

Franz Josef Och. 2003. Minimum error rate training for statistical machine translation. In *Proceedings of the 41th Annual Conference of the Association for Computational Linguistics (ACL-03)*.

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 Conference of the Association for Computational Linguistics (ACL-02)*, Philadelphia, PA.