

USING F-STRUCTURES IN MACHINE TRANSLATION EVALUATION

Karolina Owczarzak Yvette Graham
Josef van Genabith

National Centre for Language Technology
School of Computing
Dublin City University

Proceedings of the LFG07 Conference

Miriam Butt and Tracy Holloway King
(Editors)

2007

CSLI Publications

<http://csli-publications.stanford.edu/>

Abstract

Despite a growing interest in automatic evaluation methods for Machine Translation (MT) quality, most existing automatic metrics are still limited to surface comparison of translation and reference strings. In this paper we show how Lexical-Functional Grammar (LFG) labelled dependencies obtained from an automatic parse can be used to assess the quality of MT on a deeper linguistic level, giving as a result higher correlations with human judgements.

1 Introduction

The use of automatic evaluation metrics became quite widespread in the Machine Translation (MT) community, mainly because such metrics provide an inexpensive and fast way to assess translation quality. It would be highly impractical to employ humans every time MT developers wished to test whether the changes in their system are reflected in the quality of the translations, so the appearance of string-based evaluation metrics such as BLEU (Papineni et al., 2002) and NIST (Doddington, 2002) have been a great boost to the field. Both BLEU and NIST score a candidate translation on the basis of the number of n -grams shared with one or more reference translations, with NIST additionally using frequency information to weigh certain n -grams more than others. The metrics are fast to apply and intuitively easy to understand; however, these advantages come at a price. An automatic comparison of n -grams measures only the surface string similarity of the candidate translation to one or more reference strings, and will penalize any (even admissible and well-motivated) divergence from them. In effect, a candidate translation expressing the source meaning accurately and fluently will be given a low score if the lexical and syntactic choices it contains, even though perfectly legitimate, are not present in at least one of the references. Necessarily, this score would differ from a much more favourable human judgement that such a translation would receive.

The adequacy of string-based comparison methods has been questioned repeatedly within the MT community, with strong criticism for insensitivity to perfectly legitimate syntactic and lexical variation which can occur between the candidate and reference. However, almost all attempts at creating better metrics have been limited to the incorporation of local paraphrasing and/or surface reordering of elements, while ignoring structural levels of representation.

In this paper, we present a novel method that automatically evaluates the quality of translation based on the labelled dependency structure of the sentence, rather than its surface form. Dependencies abstract away from some of the particulars of the surface string (and CFG (Context-Free Grammar) tree) realization and provide a more “normalized” representation of (some) syntactic variants of a given sentence. The translation and reference files are analyzed by a treebank-based, probabilistic Lexical-Functional Grammar

(LFG) parser (Cahill et al., 2004), which produces a set of labelled dependency triples for each input. The translation set is compared to the reference set, and the number of matches is calculated, giving the precision, recall, and f-score for each particular translation.

In an experiment on 5,007 sentences of Chinese-English newswire text with associated segment-level human evaluation from the Linguistic Data Consortium's (LDC) Multiple Translation project,¹ we compare the LFG-based evaluation method with other popular metrics like BLEU, NIST, General Text Matcher (GTM) (Turian et al., 2003), Translation Error Rate (TER) (Snover et al., 2006),² and METEOR (Banerjee and Lavie, 2005), and we show that our labelled dependency representations lead to a more accurate evaluation that correlates better with human judgment. Although evaluated on a different test set, our method also outperforms the correlation with human scores reported for an earlier unlabelled dependency-based method presented in Liu and Gildea (2005).

The remainder of this paper is organized as follows: Section 2 gives a basic introduction to LFG; Section 3 describes related work; Section 4 describes our method; Section 5 gives results of two experiments on 5,007 sentences of Chinese-English newswire text from the Multiple Translation project; Section 6 discusses ongoing work; Section 7 concludes.

2 Lexical-Functional Grammar

In Lexical-Functional Grammar (Kaplan and Bresnan, 1982; Bresnan, 2001), sentence structure is represented in terms of c(onstituent)-structure and f(unctional)-structure. C-structure represents the word order of the surface string and the hierarchical organisation of phrases in terms of CFG trees. F-structures are recursive feature (or attribute-value) structures, representing abstract grammatical relations, such as SUBJ(ect), OBJ(ect), OBL(ique), ADJ(unct), etc., approximating to predicate-argument structure or simple logical forms. C-structure and f-structure are related in terms of functional annotations (attribute-value structure equations) which describe f-structures and are placed on c-structure trees.

While c-structure is sensitive to surface rearrangement of constituents, f-structure abstracts away from some of the particulars of the surface realization. The sentences *John resigned yesterday* and *Yesterday, John resigned* will receive different tree representations, but identical f-structures, shown in (1).

¹ <http://www ldc upenn edu/>

² We omit HTER (Human-Targeted Translation Error Rate), as it is not fully automatic and requires human input.

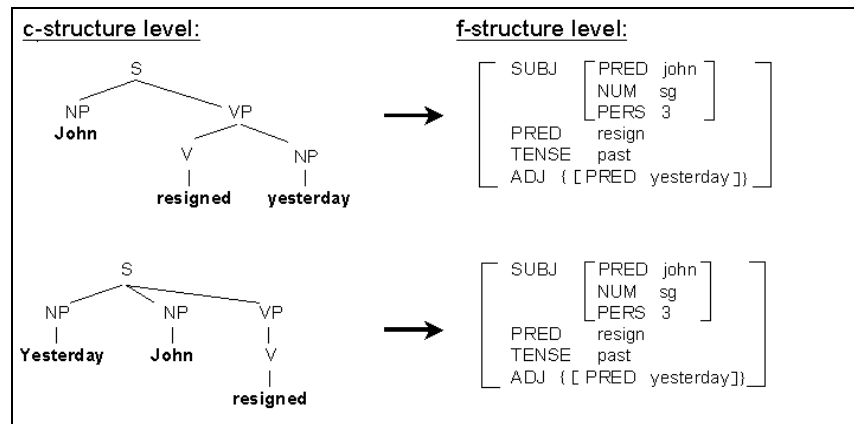


Figure 1. C-structure and f-structure

Note that if these sentences were a translation-reference pair, they would receive a less-than-appropriate score from string-based metrics. For example, BLEU with add-one smoothing³ gives this pair a score of 0.76. This is because, although all three unigrams from the “translation” (*John; resigned; yesterday*) are present in the reference (*Yesterday; John; resigned*), the “translation” contains only one bigram (*John resigned*) that matches the “reference” (*Yesterday John; John resigned*), and no matching trigrams.

The f-structure can also be described in terms of a flat set of triples. In triples format, the f-structure in (1) is represented as shown in (2). The representation in (2) is simplified in that it omits index numbers which are carried by the words (this keeps track of multiple tokens of the same lexical item in a single sentence).

```

SUBJ(resign, john)
PERS(john, 3)
NUM(john, sg)
TENSE(resign, past)
ADJ(resign, yesterday)
PERS(yesterday, 3)
NUM(yesterday, sg)

```

Figure 2. A set of dependencies in the triples format

³ We use smoothing because the original BLEU metric gives zero points to translations with fewer than one four-gram in common with the reference. We note also that BLEU is not intended for use at the segment level, but show this example for illustration only. In this example, we also ignore the punctuation in the segments to simplify things.

Cahill et al. (2004) presents a set of Penn-II Treebank-based LFG parsing resources. Their approach distinguishes 32 types of dependencies, including grammatical functions and morphological information. This set can be divided into two major groups: a group of predicate-only dependencies and a group of non-predicate (atomic) dependencies. Predicate-only dependencies are those whose path ends in a predicate-value pair, describing grammatical relations. For example, for the f-structure in (1), predicate-only dependencies would include: {SUBJ(resign, john), ADJ(resign, yesterday)}. Other predicate-only dependencies include: *apposition, complement, open complement, coordination, determiner, object, second object, oblique, second oblique, oblique agent, possessive, quantifier, relative clause, topic, and relative clause pronoun*. The remaining non-predicate dependencies are: *adjectival degree, coordination surface form, focus*, complementizer forms: *if, whether, and that, modal, number, verbal particle, participle, passive, person, pronoun surface form, tense, and infinitival clause*.

Such dependencies are often the basis of parser evaluation, where the quality of the f-structures produced automatically can be checked against a set of gold standard sentences annotated with f-structures by a linguist. The evaluation is conducted by calculating the precision and recall between the set of dependencies produced by the parser and the set of dependencies derived from the human-created f-structure. Usually, two versions of f-score are calculated: one for all the dependencies for a given input and a separate one for the subset of predicate-only dependencies.

In the experiments reported in this paper, we use the LFG parser developed by Cahill et al. (2004), which automatically annotates input text with c-structure trees and f-structure dependencies, obtaining high precision and recall rates.⁴

3 Related Research

3.1 String-Based Metrics

The insensitivity of BLEU and NIST to perfectly legitimate syntactic and lexical variation has been raised, among others, in Callison-Burch et al. (2006), but the criticism is widespread. Even the creators of BLEU point out that it may not correlate particularly well with human judgment at the sentence level (Papineni et al., 2002).

Recently a number of attempts to remedy these shortcomings have led to the development of other automatic MT evaluation metrics. Some of them concentrate mainly on allowing greater differences in word order between the translation and the reference, like General Text Matcher (Turian et al., 2003), which calculates precision and recall for translation-reference

⁴ A demo of the parser can be found here:
<http://lfg-demo.computing.dcu.ie/lfgparser.html>

pairs, weighting contiguous string matches more than non-sequential matches, or Translation Error Rate (Snover et al., 2006), which computes the number of substitutions, insertions, deletions, and shifts necessary to transform the translation text to match the reference. Others try to accommodate both syntactic and lexical differences between the candidate translation and the reference, like CDER (Leusch et al., 2006), which employs a version of edit distance for word substitution and reordering; or METEOR (Banerjee and Lavie, 2005), which uses stemming and WordNet⁵-based synonymy. Kauchak and Barzilay (2006) and Owczarzak et al. (2006) use paraphrases in conjunction with BLEU and NIST evaluation to increase the number of matches between the translation and the reference; the paraphrases are either taken from WordNet (Kauchak and Barzilay, 2006) or derived from the test set itself through automatic word and phrase alignment (Owczarzak et al., 2006). Another metric making use of synonyms is the linear regression model developed by Russo-Lassner et al. (2005), which makes use of stemming, WordNet synonymy, verb class synonymy, matching noun phrase heads, and proper name matching. Kulesza and Shieber (2004), on the other hand, train a Support Vector Machine using features such as proportion of n -gram matches and word error rate to judge a given translation's distance from human-level quality.

3.2 Dependency-Based Metrics

The metrics described in Section 3.1 use only string-based comparisons, even while taking into consideration reordering. By contrast, Liu and Gildea (2005) present three metrics that use syntactic and unlabelled dependency information. Two of these metrics are based on matching syntactic subtrees between the translation and the reference, and one is based on matching headword chains, i.e. sequences of words that correspond to a path in the *unlabelled* dependency tree of the sentence. Dependency trees are created by extracting a headword for each node of the syntactic tree, according to the rules used by the parser of Collins (1999), where every subtree represents the modifier information for its root headword. The dependency trees for the translation and the reference are converted into flat headword chains, and the number of overlapping n -grams between the translation and the reference chains is calculated. Our method, by contrast, uses *labelled* LFG dependencies, partial matching, and n -best parses, allowing us to considerably outperform Liu and Gildea's (2005) highest correlations with human judgement (they report 0.144 for the correlation with human fluency judgement and 0.202 for the correlation with human overall judgement), although it has to be kept in mind that such comparison is only tentative, as their correlation results are calculated on a different test set.

⁵ <http://wordnet.princeton.edu/>

4 LFG F-structure in MT Evaluation

As for parsing, the process underlying the evaluation of f-structure quality against a gold standard can be used in automatic MT evaluation as well: we parse the translation and the reference, and then, for each sentence, we check the set of translation dependencies against the set of reference dependencies, counting the number of matches. As a result, we obtain the precision and recall scores for the translation, and we calculate the f-score for the given pair. Because we are comparing two outputs that were produced automatically, there is a possibility that the result will not be noise-free.

To assess the amount of noise that the parser may introduce, we conducted an experiment where 100 English sentences were modified by hand in such a way that the position of adjuncts was changed, but the sentence remained grammatical and the meaning was not changed, as shown in (1).

- (1) a. We must change this system, Commissioner.
b. Commissioner, we must change this system.

This way, an ideal parser should give both the source and the modified sentence the same f-structure, similarly to the case presented in (1). The modified sentences were treated like a translation file, and the original sentences played the part of the reference. Each set was run through the parser. We evaluated the dependency triples obtained from the “translation” against the dependency triples for the “reference”, calculating the f-score, and applied other metrics (TER, METEOR, BLEU, NIST, and GTM) to the set in order to compare scores. The results, including the distinction between f-scores for all dependencies and predicate-only dependencies, are given in Table 1.

	upper bound	modified
TER	0.0	6.417
METEOR	1.0	0.9970
BLEU	1.0	0.8725
NIST	11.5232	11.1704 (96.94%)
GTM	100	99.18
dep f-score	100	96.56
dep_preds f-score	100	94.13

Table 1. Scores for sentences with reordered adjuncts

The baseline column shows the upper bound for a given metric: the score which a perfect translation, word-for-word identical to the reference, would

obtain.⁶ In the other column we list the scores that the metrics gave to the “translation” containing reordered adjuncts. As can be seen, the dependency and predicate-only dependency scores are lower than the perfect 100, reflecting the noise introduced by the parser.

To show the difference between the scoring based on LFG dependencies and other metrics in an ideal situation, we created another set of a hundred sentences with reordered adjuncts, but this time selecting only those reordered sentences that were given the same set of dependencies by the parser (in other words, we simulated having the ideal parser). As can be seen in Table 2, other metrics are still unable to tolerate legitimate variation in the position of adjuncts, because the sentence surface form differs from the reference; however, it is not treated as an error by the parser.

	upper bound	modified
TER	0.0	7.841
METEOR	1.0	0.9956
BLEU	1.0	0.8485
NIST	11.1690	10.7422 (96.18%)
GTM	100	99.35
dep f-score	100	100
dep_preds f-score	100	100

Table 2. Scores for sentences with reordered adjuncts in an ideal situation

5 Correlations with Human Judgement - MultiTrans

5.1 Experimental Design

To evaluate the correlation with human assessment, we used the data from the Linguistic Data Consortium Multiple Translation Chinese (MTC) Parts 2 and 4, which consists of multiple translations of Chinese newswire text, four human-produced references, and segment-level human evaluation scores for a subset of the translation-reference pairs. Although a single translated segment was always evaluated by more than one judge, the judges used a different reference every time, which is why we treated each translation-reference-human score triple as a separate segment. In effect, the test set created from this data contained 16,800 segments. We randomly selected 5,007 segments as our test set, while the remaining segments served as a training corpus for those versions of our test method that required the training

⁶ Two things have to be noted here: (1) in case of NIST the perfect score differs from text to text, which is why we provide the percentage points as well, and (2) in case of TER the lower the score, the better the translation, so the perfect translation will receive 0, and there is no bound on the score, which makes this particular metric extremely difficult to directly compare with others.

of weights. As in the previous experiment, the translation was scored using BLEU, NIST, GTM, TER, METEOR, and our labelled dependency-based method.

5.2 Labelled Dependency-Based Method

The results, presented in Table 3, show that although the basic labelled dependency-based evaluation method achieves a high correlation with human scores for translation fluency, it is only average in its correlation with human judgement of translation accuracy, falling short of some string-based metrics. This suggests that the dependency f-score, at least as calculated in the evaluation method used for parsing, might not be the ideal reflection of the true quality of the translation. This could be due to the dependency triple f-score assigning equal weight to each dependency triple. For parser evaluation this is appropriate, but for MT evaluation it may not be. Since the task of automatic MT evaluation attempts to replicate human judgments of a given candidate translation for adequacy and fluency, the type of relation that the dependency encodes may influence its importance in the evaluation.

H_FL		H_AC		H_AV	
GTM	0.172	METEOR	0.278	METEOR	0.242
dep	0.161	NIST	0.273	NIST	0.238
BLEU	0.155	dep	0.256	dep	0.235
METEOR	0.149	dep_preds	0.240	dep_preds	0.216
NIST	0.146	GTM	0.203	GTM	0.208
dep_preds	0.143	BLEU	0.199	BLEU	0.197
TER	0.133	TER	0.192	TER	0.182

Table 3: Pearson’s correlation between human scores and evaluation metrics. Legend: **dep** = dependency-based method, **_preds** = predicate-only, **M** = METEOR, **H_FL** = human fluency score, **H_AC** = human accuracy score, **H_AV** = human average score.

For example, predicate-only dependencies (like SUBJ, OBJ, ADJ, etc.) encode a specific relation between two items, and only when both of these items happen to occur in that specific labelled dependency relation is the dependency counted as a match against the reference. This proves problematic when using dependencies to evaluate MT output, since we might encounter lexical variation: in a candidate-reference pair *John quit yesterday* and *John resigned yesterday* none of the predicate-only dependencies will match, e.g. candidate: {SUBJ(quit, John), ADJ(quit, yesterday)}, reference: {SUBJ(resign, John), ADJ(resign, yesterday)}. The predicate-only score would therefore be zero. However, if we allow partial matches for predicate-only dependencies, this should accommodate cases where an object might find itself in the correct relation, but with an incorrect partner. This modified method would give us an f-score of 0.5 (candidate: {SUBJ(quit,_),

SUBJ(_ ,John), ADJ(quit,_) , ADJ(_ ,yesterday)}; reference: {SUBJ(resign,_) , SUBJ(_ ,John), ADJ(resign,_) , ADJ(_ , yesterday)}).

Another problem stemming from the equal treatment of all dependencies is that lexical items and their resulting grammatical categories naturally differ with respect to how many atomic (non-predicate) dependencies they generate. For example, a noun phrase like *the chairman* generates three atomic dependencies from its atomic features PERS, NUM and DET, whereas a verb like *resign* might generate only a single atomic dependency for its TENSE feature. As a result, the f-score for the overall dependency triples match implicitly weights the words in the sentence by the number of atomic features the word receives at f-structure level. For example, if an MT system incorrectly translates the noun *chairman*, it affects the final score three times as much as an incorrect translation of the word *resign*. Individual lexical items can easily be given an even influence on the final score by assigning each an equal weight in the overall score, irrespective of the number of dependency relations they generate. This means that a partial f-score is calculated at the lexical item level from all the dependencies relating to this item, and then all the partial f-scores are averaged at the segment level to give the final f-score for the segment.

In addition to this, the information encoded in predicate-only dependencies and atomic feature-value pairs could relate to human judgments of translation quality differently. We investigated this by calculating a score for the atomic features only and a separate score for the predicate-only triples and combining the two scores using automatically optimized weights.

We implemented a number of ways in which predicate and atomic dependencies combine in order to arrive at the final sentence-level f-score, and we calculated the correlation between each of these combinations and human assessment of translation quality. The results of these modifications are presented in Table 4. Interestingly, all the improved f-score calculations raise the correlation with human MT evaluation scores over the values displayed by the original f-score calculation; the only scores showing lower correlation than the traditional method are partial f-scores for predicates-only and atomic-features-only. It is also important to note that this increase in correlation, even if not enough to outperform the highest-ranking string-based metrics in the areas of human fluency and accuracy judgement (GTM and METEOR, respectively), is nevertheless enough to place one of the dependency-based f-score calculations (partial match for predicate dependencies plus all non-grouped atomic dependencies) at the top of the ranking when it comes to the general correlation with the average human score (which combines fluency and accuracy).

Method	H_FL	Method	H_AC	Method	H_AV
p+a(g)	0.1653	pm+a	0.2666	pm+a	0.2431
pm+a	0.1648	w_pm+w_a(g)	0.2648	w_pm+w_a(g)	0.2415
pm+a(g)	0.1648	pm+a(g)	0.2631	pm+a(g)	0.2409
w_pm+w_a(g)	0.1641	w_p+w_a(g)	0.2560	p+a(g)	0.2360
w_p+w_a(g)	0.1631	a(g)	0.2560	w_p+w_a(g)	0.2352
original	0.1613	original	0.2557	a(g)	0.2348
a(g)	0.1610	p+a(g)	0.2547	original	0.2347
pm	0.1579	pm	0.2479	pm	0.2283
p	0.1427	p	0.2405	p	0.2165

Table 4: Pearson’s correlation between human scores and variations of f-score dependency scores. Types of dependencies: p = predicate, pm = partial match for predicate, a = atomic, a(g) = atomic grouped by predicate, w_ = optimally weighted, original = basic f-score, H_FL = human fluency score, H_AC = human accuracy score, H_AV = human average score.

Note also that almost all versions of our method show higher correlations than the results reported in Liu and Gildea (2005): 0.144 for the correlation with human fluency judgement, 0.202 for the correlation with human overall judgement, with the proviso that the correlations are calculated on a different test set.

6 Current and Future Work

Fluency and accuracy are two very different aspects of translation quality, each with its own set of conditions along which the input is evaluated. Therefore, it seems unfair to expect a single automatic metric to correlate highly with human judgements of both fluency and accuracy at the same time. This pattern is very noticeable in Table 3: if a metric is (relatively) good at correlating with fluency, its accuracy correlation suffers (GTM might serve as an example here), and the opposite holds as well (see METEOR’s scores). It does not mean that any improvement that increases the method’s correlation with one aspect will result in a decrease in the correlation with the other aspect; but it does suggest that a possible direction for development would be to target these correlations separately, if we want our automated metrics to reflect human scores better. At the same time, string-based metrics might have already exhausted their potential when it comes to increasing their correlation with human evaluation; as has been pointed out before, these metrics can only tell us that two strings differ, but they cannot distinguish legitimate grammatical variance from ungrammatical variance. As the quality of MT improves, the community will need metrics that are more sensitive in this respect. After all, the true quality of MT depends on producing grammatical output which describes the same concepts (or proposition) as the source utterance, and the string identity with a reference is only a very arbitrary approximation of this goal.

In order to maximize the correlation with human scores of fluency, we plan to look more closely at the parser output, and implement some basic transformations which would allow an even deeper logical analysis of input (e.g. passive to active voice transformation).

As to the correlations with human judgments of accuracy, we found that adding WordNet synonyms to the matching process increases the scores. The use of synonyms in matching allows us to account for legitimate lexical variation that can occur between the translation and the reference. For example, if our “translation” in Figure 1 *John resigned yesterday* contained the verb *quit* instead of *resign*, the number of matches would decrease even though a human judge would be able to recognize the equivalent meaning; however, if we automatically search WordNet synonym sets and find that *quit* and *resign* are in fact synonyms, we can still count the match. Results of these experiments are presented in Owczarzak et al. (2007a,b).

7 Conclusions

In this paper we present a novel way of evaluating MT output. So far, most metrics have relied on comparing translation and reference on a string level. Even given reordering, stemming, and synonyms for individual words, current methods are still far from reaching human ability to assess the quality of translation. Our method compares the sentences on the level of their grammatical structure, as exemplified by their f-structure labelled dependency triples produced by an LFG parser. The labelled dependency-based method can be further augmented by allowing partial matching for predicate dependencies or WordNet synonyms. In our experiments we showed that one version of the dependency-based method correlates higher than any other metric with the average human score. The use of labelled dependencies in MT evaluation is a rather new idea and requires more research to improve it, but the method shows potential to become an accurate, yet automatic, evaluation metric.

Acknowledgements

This work was partly funded by Microsoft Ireland PhD studentship 2006-8 for the first author of the paper and Science Foundation Ireland studentship P07077-60101 for the second author. We would also like to thank our reviewers for their insightful comments. All remaining errors are our own.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation*

Measures for MT and/or Summarization at the Association for Computational Linguistics Conference 2005: 65-73. Ann Arbor, Michigan.

Joan Bresnan. 2001. *Lexical-Functional Syntax*, Blackwell, Oxford.

Aoife Cahill, Michael Burke, Ruth O'Donovan, Josef van Genabith, and Andy Way. 2004. Long-Distance Dependency Resolution in Automatically Acquired Wide-Coverage PCFG-Based LFG Approximations, In *Proceedings of Association for Computational Linguistics 2004*: 320-327. Barcelona, Spain.

Chris Callison-Burch, Miles Osborne and Philipp Koehn. 2006. Re-evaluating the role of BLEU in Machine Translation Research. *Proceedings of the European Chapter of the Association for Computational Linguistics 2006*: 249-256. Oslo, Norway.

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

George Doddington. 2002. Automatic Evaluation of MT Quality using N-gram Co-occurrence Statistics. *Proceedings of Human Language Technology Conference 2002*: 138-145. San Diego, California.

Ronald M. Kaplan and Joan Bresnan. 1982. *Lexical-functional Grammar: A Formal System for Grammatical Representation*. In J. Bresnan (ed.), *The Mental Representation of Grammatical Relations*. MIT Press, Cambridge.

David Kauchak and Regina Barzilay. 2006. Paraphrasing for Automatic Evaluation. *Proceedings of Human Language Technology – North American Chapter of the Association for Computational Linguistics Conference 2006*: 45-462. New York, New York.

Alex Kulesza and Stuart M. Shieber. 2004. A learning approach to improving sentence-level MT evaluation. In *Proceedings of the Conference on Theoretical and Methodological Issues in Machine Translation 2004*: 75-84. Baltimore, Maryland.

Gregor Leusch, Nicola Ueffing and Hermann Ney. 2006. CDER: Efficient MT Evaluation Using Block Movements. *Proceedings of European Chapter of the Association for Computational Linguistics Conference 2006*: 241-248. Trento, Italy.

Ding Liu and Daniel Gildea. 2005. Syntactic Features for Evaluation of Machine Translation. In *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization at the Association for Computational Linguistics Conference 2005*. Ann Arbor, Michigan.

Karolina Owczarzak, Declan Groves, Josef van Genabith, and Andy Way. 2006. Contextual Bitext-Derived Paraphrases in Automatic MT

Evaluation. *Proceedings of the Workshop on Statistical Machine Translation at the Human Language Technology – North American Chapter of the Association for Computational Linguistics Conference 2006*: 86-93. New York, New York.

Karolina Owczarzak, Josef van Genabith, and Andy Way. 2007a. Dependency-Based Automatic Evaluation for Machine Translation. *Proceedings of the HLT-NAACL 2007 Workshop on Syntax and Structure in Statistical Machine Translation*: 86-93. Rochester, New York.

Karolina Owczarzak, Josef van Genabith, and Andy Way. 2007b. Labelled Dependencies in Machine Translation Evaluation. *Proceedings of the ACL 2007 Workshop on Statistical Machine Translation*: 104-111. Prague, Czech Republic.

Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of Association for Computational Linguistics Conference 2002*: 311-318. Philadelphia, Pennsylvania.

Grazia Russo-Lassner, Jimmy Lin, and Philip Resnik. 2005. A Paraphrase-based Approach to Machine Translation Evaluation. Technical Report LAMP-TR-125/CS-TR-4754/UMIACS-TR-2005-57, University of Maryland, College Park, Maryland.

Mathew Snover, Bonnie Dorr, Richard Schwartz, John Makhoul, Linnea Micciula. 2006. A Study of Translation Error Rate with Targeted Human Annotation. *Proceedings of the Association for Machine Translation in the Americas Conference 2006*: 223-231. Boston, Massachusetts.

Joseph P. Turian, Luke Shen, and I. Dan Melamed. 2003. Evaluation of Machine Translation and Its Evaluation. *Proceedings of MT Summit 2003*: 386-393. New Orleans, Louisiana.