Multiword Expressions in the Context of Statistical Machine Translation

Mahmoud Ghoneim
Center for Computational Learning Systems
Columbia University
475 Riverside Drive MC 7717
New York, NY 10115
mah.ghoneim@gmail.com

Mona Diab
Department of Computer Science
George Washington University
801 22nd Street NW
Washington DC 20052
mtdiab@email.gwu.edu

Abstract

Incorporating semantic information in the Statistical Machine Translation (SMT) framework is starting to gain some popularity in both the semantics and translation communities. In this paper, we present encouraging results obtained from experiments conducted on English to Arabic SMT system using static, dynamic, and hybrid integration of fine-grained Multiword Expression (MWE). We achieve an improvement up to 0.82 absolute BLEU score by integrating MWEs over a vanilla SMT system. We empirically show that different MWE types require different integration methods in the SMT framework.

1 Introduction

Multiword expressions (MWEs) are roughly defined by (Sag et al., 2002) as "idiosyncratic concepts that cross word boundaries (spaces)." MWEs are widely used, 41% of the entries in WordNet 1.7 (Fellbaum, 1998) are MWEs, but unfortunately they have proved to be hard to model in natural language processing applications. Typical statistical machine translation (SMT) systems, in particular, do not explicitly model MWEs. This might indicate that state of the art SMT systems are doing well without having any knowledge of whether a given phrase is a multiword expression or not. However, recent research (Carpuat and Diab 2010, Bouamor et al., 2012) show that explicitly modeling MWEs in the SMT framework yields non-negligible gains depending on the integration method.

In this paper we study explicit modeling of the diverse kinds of MWEs in a phrase-based SMT framework for the English-Arabic language pair. This paper is organized as follows: section 2 overviews the different types of MWEs, section 3 reviews the previous work related to MWEs and SMT. Section 4 details our approach followed by the results in section 5. Our discussion of the results is presented in section 6 and finally the conclusions are in section 7.

2 Multiword Expressions Classification

According to (Sag et al., 2002), MWEs are broadly classified into institutionalized phrases and lexicalized phrases based on the varying degree of lexical rigidity and semantic compositionality.

Institutionalized phrases are conventionalized phrases that are syntactically and semantically compositional, but statistically idiosyncratic (e.g. "traffic light", "to kindle excitement".)

Lexicalized phrases have at least in part idiosyncratic syntax or semantics. They can be further broken down into:

(a) Fixed expressions which undergo neither morphosyntactic variation, nor internal modification (e.g. "by and large", "every which way") [AV, AJ],

(b) Semi-fixed expressions such as (1) non-decomposable idioms (e.g. "kick the bucket") [VNC], (2) compound nominal (e.g. "car park", "part of speech") [NNC], and (3) proper names and named entities (e.g. "New York") [NE].

(c) Syntactically-flexible expressions such as (1) verb particle construction (e.g. "write up", "look up") [VPC], (2) light verb constructions (e.g. "make a decision") [LVC], and (3) decomposable idioms (e.g. "sweep under the rug") [VNC].

3 Related Work

Previous work has focused on automatically learning and integrating translations of very specific MWE categories, such as, for instance, idiomatic Chinese four character expressions (Bai

International Joint Conference on Natural Language Processing, pages 1181–1187,
Nagoya, Japan, 14-18 October 2013.

et al., 2009.) MWEs have also been defined not from a lexical semantics perspective but from a SMT error reduction perspective, as phrases that are hard to align during SMT training (Lambert and Banchs, 2005). For each of these particular cases, translation quality improved by augmenting the SMT translation lexicon with the learned bilingual MWEs either directly or through improved word alignments.

Ren et al. (2009) described a method integrating an in-domain bilingual MWE to Moses by introducing an additional feature that identifies whether or not a bilingual phrase contains bilingual MWEs. This approach was generalized in Carpuat and Diab (2010) who replaced the binary feature by a count feature representing the number of MWEs in the source language phrase. They present results on a large data set of English to Arabic SMT. They introduce two ways of integrating MWE knowledge in the SMT framework: Static and Dynamic integration. For Static integration, MWE tokens in the source data are grouped together with an underscore. While in Dynamic integration, the MWEs are identified in the phrase table and an additional weighted feature, as a soft constraint, is added to the phrase translation table. Carpuat and Diab (2010) focus only on MWEs as identified in WordNet (Fellbaum, 1998) with no explicit distinction between the different types of MWEs. Accordingly, the MWEs are considered a single type with no attention to various POS information. Our work here is taking a much fine grained approach and deeper study and analysis.

4 Approach

We adopt a Phrase-based SMT framework, Moses (Koehn et al., 2007). In the following subsections, we address the issue of representation of MWE in our SMT pipeline and then we investigate the manner in which the MWE information is integrated in the SMT framework.

4.1 Data Sets

For training the translation models, we use LDC GALE newswire parallel Arabic-English corpus (LDC2007E103) (a total of 474299 sentence pairs / about 10M un-tokenized words / 12M tokenized words). The Log-Linear model features weights are tuned using the newswire part of NIST MT06 (765 sentence pairs) as the tuning dataset and BLEU (Papineni et al., 2002) as the objective function. For training the language model (LM), we use the LDC Arabic GIGAWORD 4th edition (LDC2009T30) (about 850M un-tokenized words).

We use the newswire part of NIST-MT04 (707 sentences) as our development test-set to compare performance and select combinations of different conditions. We report results using two blind test-sets; NIST-MT05 (1056 sentences) and the newswire part of NIST-MT08 (813 sentences). These standard test sets are originally designed to test Arabic to English translation systems thus it consists of one Arabic source set and four English human reference translation sets. To use these test sets for testing English to Arabic translation systems, we created new test sets where the source set is constructed by concatenating the four English human translations of the original standard test set, and the reference set is constructed by duplicating the original standard test set Arabic source four times. This means that the new test sets have four times the number of sentences of the original standard test sets. Increasing the test set size enhances the reliability of the evaluation scores as reported by (Zhang and Vogel 2010).

4.2 MWEs lists

We need a mechanism by which to identify MWE in the source English text. We rely on two identification sources depending on the type of MWE: an MWE list extracted from a wide coverage lexical database and a named entity recognition tool. As mentioned earlier in section 2, we consider several types of MWEs for this study: Verb-based MWEs (VNC, VPC, and LVC), Noun-based MWEs (NNC, and NE), Adjective (AJ) and Adverb (AV) based MWE.

WordNet Extracted MWEs Lists:

For the VPC, VNC, LVC, NNC and AJ and AV categories of MWE, we extract an extensive list from the wide coverage English WordNet database 3.0. (Fellbaum,1998). Table 1 shows the number of MWEs extracted from WordNet 3.0 dictionaries. It is worth noting that the MWE.V list comprises all three types of verbal MWEs (VNC, VPC, LVC), moreover the MWE.N includes NNC and some NEs as listed in WordNet.

MWE list	# MWE types
MWE.V	3,089
MWE.N	62,244
MWE.AJ	3,358
MWE.AV	826

Table 1: WordNet 3.0 based MWE statistics

Named Entities Tagging:

We consider Named Entities (NEs) as another type of MWE. To construct our NEs list, we exploit a named entity tagger, the Stanford NER [SNER] (Finkel et al., 2005). SNER tags named entities in a given English text into three categories: 1) Person 2) Organization and 3) Location. We are interested in Multiword NEs only and pay no attention to the different NE categories. The extracted NEs list consists of the 65616 Multiword NEs tagged by SNER in our training corpus.

There are some overlaps between the NEs list and the MWE lists extracted from WordNet as shown in table 2. The large overlap is between the NEs list and the MWE.N, which contains NEs as listed in WordNet 3.0.

	MWE.N	MWE.AJ	MWE.AV
# types	1216	24	5
Examples	abraham lincoln abu dhabi abu sayyaf adam smith addis ababa adriatic sea	african american anti american central american costa rican east african eastern orthodox	north east north northeast north west south east south west

Table 2: Overlaps between the WordNet MWEs lists and the NEs list

Matching Algorithm:

In order to identify the MWE in the source English side of the parallel data, we use a Maximum Forward Matching algorithm that finds the longest matching MWE in the text. The algorithm matches over the tokenized version of the data and if no match, it backs-off to the lemmatized version to account for the different inflectional forms of the MWE (e.g. “take place” and “took place”.) Our current matching algorithm doesn’t handle gap flexibility like in the phrasal verbs MWEs (i.e. “break up” is handled while “break it up” is not.)

4.3 SMT System

Data preprocessing and models generation:

The Arabic side of the train, tune, development and test data sets and the language model training data sets are tokenized using AMIRA 2.1 toolkit (Diab 2009, Diab et al., 2007) into the Arabic TreeBank tokenization scheme. The Arabic side of the training data is further processed to generate a lemmatized version used in the alignment stage of the SMT pipeline. We use the undiacritized version (both tokenized and lemmatized) in all our experiments.

The English side is tokenized using Tree Tagger (Schmid, 1994). It is then tagged using the selected MWE list according to the condition under investigation. The English lemmatized version of the training data is also generated for use in alignment.

We used SRILM toolkit (Stolcke, 2002) to create a 5-gram Arabic LM modified using Kneser-Ney smoothing.

In all our experimental conditions, the parallel corpus is word-aligned using GIZA++ in both translation directions using the lemmatized version of both sides to decreases data sparseness, and phrase translations of up to 10 words are extracted from the tokenized version of both sides using the grow-diag-final-and heuristic (Koehn et al., 2007).

We optimized log-linear model feature weights using Minimum Error Rate Training (MERT) (Och, 2003). To account for the instability of MERT, we run the tuning step three times per condition with different random seeds and use the optimized weights that give the median score.

Integration Methods:

(a) Static Integration (S)

In Static integration of MWEs in SMT, MWEs in English training, tuning and testing data are underscored as a preprocessing step based on a pattern match to the WN list entries and NER results. Hence static integration is a manipulation on the data representation, the SMT system is kept intact.

(b) Dynamic Integration (D)

Dynamic integration is a soft constraint strategy that adds a new feature into the log linear model of phrase-based SMT. It is a count feature indicating the number of MWEs in the English phrase in the phrase table, thereby biasing the system, at decoding time, towards using phrases that do not break MWEs. The training, tuning, development and test data do not undergo any MWEs annotation (no underscoring).

(c) Zone Integration (Z)

We define constrained reordering zones for all MWEs found in the test data and the decoder is forced to respect these boundaries while constructing the translation hypothesis. This is easily represented using XML tags in the system input to Moses decoder (Koehn and Haddow, 2009). It is worth noting that words within a zone are not necessarily translated as a single phrase and can be reordered; input phrases that cross zone boundaries can be used in translation hypotheses without breaking the reordering constraint.

(d) Hybrid Integration

Motivated by the development-set results of the previous integration methods and MWEs schemes, we carried out a set of experiments investigating combining the best performing conditions.

MWEs Schemes:

We created 7 MWEs schemes combining the various types of WordNet-based MWE lists and NEs list. They are listed in Table 3, along with the number of types and tokens of MWEs found in the training data according to each of the MWE Schemes.

We combine MWEs schemes and integration methods to get the different experimental conditions listed in Table 4. Here is some example input preprocessing for the same sentence according to different conditions:

-Baseline (and all dynamic integration):

invading iraqis kurdistan is no longer an easy task .

-S_VAA[1]*:*

invading iraqis kurdistan is no_longer an easy task .

-S_NN:

invading iraqis_kurdistan is no longer an easy task .

-Z_VAA+NN:

invading <zone> iraqis kurdistan </zone> is <zone> no longer </zone> an easy task .

5 Evaluation Results

We used four standard MT metrics[2]; BLEU (Papineni et al., 2002), NIST (Doddington, 2002), METEOR[3] (Banerjee and Lavie, 2005), and TER (Snover et al., 2006), to report and compare performance of different experimental conditions. Table 4, summarizes the results.

The results show that, for the three integration methods (S, D and Z), the only conditions that help across all test-sets are S_VAA and D_VAA. S_VAA gives the best results except for METEOR where D_NN and D_NE are outperforming S_VAA.

We want to investigate which part of the SMT pipeline does S_VAA condition help, so we carried another experiment (A_VAA) where the VAA is used in the alignment stage of the pipeline only. We simply removed underscores from the input phrases in the phrase table and the lexical reordering table and used the new tables as A_VAA tables. The tune and test data-sets are the same as the normal baseline (no underscoring). The results show that the major part of the S_VAA configuration enhancement is actually coming from the alignment stage.

Motivated by the development set results of S_VAA, A_VAA and the enhancement of METEOR scores by D_NN and D_NE, we carried out a couple of experiments investigating hybrids of the integration methods.

S_VAA-*: In these configurations we use static integration for VAA and dynamic integration for NE and/or NN. For example, for S_VAA-D_NE the input phrases in the phrase table have VAA MWEs underscored and the probabilities have the added extra feature counting NEs in the input phrase. The train, tune and test data for this configuration has VAA MWEs underscored.

A_VAA-*: In these configurations we use the phrase tables of the S_VAA and remove underscores from the input phrases. We then add the extra feature indicating the counts of the NE and/or NN MWEs found in the input phrase.

Table 4 shows that A_VAA-D_NE+NN gives the best overall consistent performance with absolute BLEU score improvement of 0.63 for MT04-NW, 0.82 for MT05 and 0.45 for MT08-NW.

6 Discussion

Static integration mainly helps when the MWE is a fixed expression (AV, AJ) that needs to be translated as a whole non-compositionally. That's why we see the VAA condition (more than half of its list is fixed MWEs) giving the best results. Static integration also helps for semi-fixed expressions (VNC, NNC and NEs) conditioned by having enough training samples otherwise we increase OOV. If we look at table 3, we can see that the average number of tokens per type for all NE conditions is very low. This is mainly due to the huge number of NE types. That's why NEs schemes do not show any improvement using static integration. On the other hand, S_NN shows some inconsistent improvements depending on the data sparsity. For example, in our sample test sentence, S_NN condition

[1] We use the convention: IntegrationMethod_MWEScheme [-IntegrationMethod_MWEScheme]* to label different conditions: e.g "S_VAA-D_NE+NN" refers to a hybrid integration where the "VAA" MWEs are statically integrated and the "NE+NN" MWEs are dynamically integrated.

[2] We report case-sensitive scores as our system output is in Buckwalter transliteration.

[3] For METEOR scores, we used "exact" module only.

created the new token “iraqis_kurdistan” which is not in the training data.

Dynamic integration helps solving this data sparsity issue by introducing a new feature that is weighted globally using all evidences belonging to the same category to favor phrase pairs with unbroken MWE of that category. That’s why we see some improvements for NEs and NNs in addition to VAA.

MWE Scheme	MWE List	#Lemma Types	# Token Types	# Tokens	(Tokens/ Types)
NN	MWE.N	8,075	10,503	329,116	31.33
VAA	MWE.V, MWE.AJ , MWE.AV	3,003	5,733	184,899	32.25
VAA+NN	VAA, MWE.N	10,698	15,571	494,528	31.76
NE	NE	65,634	65,616	290,564	4.43
NE +NN	MWE.N, NE	72,308	74,674	502,782	6.73
NE+VAA	VAA, NE	68,600	71,308	472,718	6.63
NE+VAA+NN	VAA+NN, NE	74,915	79,728	667,686	8.37

Table 3. MWEs Schemes Statistics

Experiments	Development Set MT04-NW				Blind Test Set MT05				Blind Test Set MT08-NW			
	BLEU	NIST	MET	TER	BLEU	NIST	MET	TER	BLEU	NIST	MET	TER
Baseline	41.28	8.24	59.58	44.43	38.65	8.17	56.60	47.49	33.82	7.45	53.51	53.84
S_NE	37.54	7.57	56.59	46.99	35.86	7.56	54.08	49.67	31.57	7.00	51.64	55.49
S_NE+NN	36.67	7.39	56.82	48.23	35.05	7.39	54.49	50.78	30.37	6.79	51.66	57.23
S_NE+VAA	37.90	7.62	56.48	46.41	36.31	7.61	54.05	49.02	32.10	7.07	51.69	54.51
S_NE+VAA+NN	37.85	7.57	56.49	46.81	35.80	7.55	53.89	49.59	31.28	6.96	51.32	55.62
S_NN	40.87	8.12	59.74	45.13	38.90	8.11	57.07	47.82	33.26	7.33	53.59	54.71
S_VAA	41.82	8.30	59.77	44.01	39.47	8.27	57.14	46.71	33.99	7.51	53.76	53.28
S_VAA+NN	41.16	8.17	59.69	44.56	38.94	8.12	56.98	47.42	33.12	7.33	53.45	54.44
D_NE	41.07	8.16	**60.05**	44.74	38.89	8.12	57.18	47.57	33.33	7.36	53.85	54.34
D_NE+NN	40.86	8.16	59.69	44.78	38.74	8.11	56.94	47.63	33.56	7.38	53.72	54.18
D_NE+VAA	40.80	8.15	59.56	44.91	38.83	8.11	56.96	47.70	33.37	7.36	53.62	54.35
D_NE+VAA+NN	41.00	8.16	59.50	44.59	39.04	8.14	56.88	47.30	33.72	7.40	53.66	53.81
D_NN	41.33	8.20	**60.05**	44.45	39.20	8.15	57.27	47.29	33.66	7.39	53.73	54.10
D_VAA	41.36	8.24	59.64	44.33	38.83	8.18	56.72	47.33	33.94	7.46	53.55	53.76
D_VAA+NN	41.12	8.20	59.70	44.58	39.06	8.17	57.02	47.29	33.66	7.41	53.69	53.99
Z_NE	41.15	8.23	59.48	44.53	38.61	8.16	56.57	47.53	33.83	7.45	53.52	53.81
Z_NE+NN	41.12	8.23	59.49	44.54	38.59	8.16	56.56	47.53	33.82	7.45	53.52	53.80
Z_NE+VAA	41.13	8.23	59.45	44.53	38.60	8.16	56.57	47.52	33.78	7.45	53.51	53.83
Z_NE+VAA+NN	41.11	8.23	59.46	44.53	38.60	8.16	56.56	47.53	33.78	7.45	53.50	53.82
Z_NN	41.25	8.24	59.59	44.43	38.61	8.16	56.58	47.50	33.80	7.44	53.49	53.85
Z_VAA	41.24	8.24	59.53	44.43	38.64	8.17	56.59	47.48	33.78	7.45	53.51	53.85
Z_VAA+NN	41.22	8.24	59.54	44.43	38.62	8.16	56.58	47.49	33.76	7.44	53.49	53.86
A_VAA	41.43	8.22	59.96	44.29	39.66	8.21	**57.42**	46.85	33.96	7.45	54.03	53.54
A_VAA-D_NE	41.85	8.29	59.95	43.80	**39.73**	8.28	57.34	46.50	34.15	7.50	53.86	53.09
A_VAA-D_NE+NN	**41.91**	**8.37**	59.63	**43.38**	39.47	**8.35**	57.08	**46.03**	**34.27**	**7.61**	53.78	**52.43**
A_VAA-D_NN	41.63	8.25	59.79	44.16	39.64	8.25	57.29	46.73	34.16	7.49	**54.12**	53.24
S_VAA-D_NE	40.79	8.14	59.58	44.91	39.11	8.15	57.19	47.39	33.44	7.38	53.91	54.02
S_VAA-D_NE+NN	41.78	8.28	59.60	43.80	39.46	8.26	57.03	46.48	34.21	7.49	53.69	52.91
S_VAA-D_NN	41.41	8.22	59.68	44.30	39.66	8.22	57.34	46.89	33.83	7.44	53.69	53.58

Table 4. BLEU,NIST, METEOR and TER scores of the different experimental conditions for NIST test sets MT04-NW, MT05 and MT08-NW*[4]

[4] The gray highlighted cells indicate enhancement over Baseline. The Bold underlined score per column is the best score for that Testset/Metric. (Note: lower TER scores indicate better performance)

Zone integration is not helping (except non-significantly[5] for NEs on MT08-NW), this is due to the fact that marking MWEs as zones and enforcing decoder to respect these zones does not prevent the decoder from translating MWEs compositionally. While the decoder is not allowed to translate out of zone phrases unless it fully finishes translating the words in the zone, it is permissible to divide the zone into any combination of phrases and translate these phrases individually and in any order.

Following are the translation of our sample test sentence for selected conditions:

-Ref:

vm An gzw krdstAn AlErAq lm yEd mhmp shlp .

-Baseline:

gzw ErAqy krdstAn lm yEd shlA .

-S_VAA:

gzw ErAqy krdstAn lys mhmp shlp .

-S_NN:

gzw iraqis_kurdistan lm yEd shlA .

-S_VAA+NN:

gzw iraqis_kurdistan lm tEd mhmp shlp .

-Z_VAA+NN:

gzw ErAqy krdstAn lm yEd shlA .

7 Conclusion

Our study indicates that, at least for our language pair, different MWE types require different integration methods in the SMT pipeline where the more flexible an MWE is, the more the dynamic the integration needs to be. Therefore, for NE and NN, dynamic integration yields the best results. While for VAA, which tend to be more rigid, we gain the most from static integration.

Our results strongly suggest that explicit modeling for MWE and their various types definitely impact SMT performance positively. This is important since the number of MWE (VAA+NN+NE) tokens in the text only amounts to a total of 5.3% of the data, even though in terms of type ratio, MWEs (VAA+NN+NE) account for 46% of the types (indicating that we see a lot of variability in type but with very low frequency), yet we see gains of up to 0.82 absolute BLEU points (for A_VAA-D_NE+NN MT05). We anticipate such effects to be even more pronounced in other more nuanced data sets such as blogs and broadcast conversations where the use of MWEs is pervasive compared to Newswire.

For future work, we plan to extend our matching algorithm to account for syntactically flexible MWEs by allowing gaps within MWE. We also plan to enhance feature engineering of the dynamic integration by assigning each MWE type a dedicated feature in the model. Finally we plan to extend our study to different language pairs and for MWEs in both source and target languages.

Acknowledgments

We would like to thank the feedback provided by three anonymous reviewers.

This work was partially supported by the DARPA BOLT program.

References

Bai, M.H., You, J.M., Chen, K.J., and Chang, J. 2009. *Acquiring translation equivalences of multiword expressions by normalized correlation frequencies*. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, 478–486.

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, Ann Arbor, MI, USA, 65–73.

Bouamor, D., Semmar, N., Zweigenbaum, P. 2012. *Identifying bilingual Multi-Word Expressions for Statistical Machine Translation*. In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey

Carpuat, M., and Diab, M. 2010. *Task-based evaluation of multiword expressions: a pilot study in statistical machine translation*. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT '10). Association for Computational Linguistics, Stroudsburg, PA, USA, 242-245.

Diab, M. 2009. *Second Generation Tools (AMIRA 2.0): Fast and Robust Tokenization, POS tagging, and Base Phrase Chunking*. MEDAR 2nd International Conference on Arabic Language Resources and Tools, April, Cairo, Egypt

Diab, M., Hacioglu, K., and Jurafsky, D. 2007. *Automatic Processing of Modern Standard Arabic Text.* In Arabic Computational Morphology: Knowledge-based and Empirical Methods, A. Soudi, A. Bosch and G. Neumann, Springer, The Netherlands, 159-180.

[5] Statistical significance tests use bootstrapping methods as detailed in (Zhang and Vogel, 2010)

Doddington, G. 2002. *Automatic evaluation of MT quality using n-gram co-occurrence statistics*. In Proceedings of Human Language Technology Conference 2002, San Diego, CA, USA, 138–145.

Fellbaum, C. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Finkel, J., Grenager, T., and Manning C. 2005. *Incorporating non-local information into information extraction systems by Gibbs sampling*. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL '05). Association for Computational Linguistics, Stroudsburg, PA, USA, 363-370.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantine, A., and Herbst, E. 2007. *Moses: open source toolkit for statistical machine translation*. In Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions (ACL '07). Association for Computational Linguistics, Stroudsburg, PA, USA, 177-180.

Lambert, P., and Banchs, R. 2005. *Data inferred multiword expressions for statistical machine translation*. In Proceedings of Machine Translation Summit X, Phuket, Thailand, 396–403.

Och, F., 2003. *Minimum error rate training for statistical machine translation*. In Proceedings of the 41th Annual Meeting of the Association for Computational Linguistics (ACL), Sapporo, July

Pal, S., Supin, K.N., Pavel, P., Sivaji, B., and Way, A. 2010. *Handling Named Entities and Compound Verbs in Phrase-Based Statistical Machine Translation*. In Proceedings of the 2010 Workshop on Multiword Expressions: from Theory to Applications (MWE 2010). Association for Computational Linguistics, Stroudsburg, PA, USA, 45-53.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.J. 2002. *BLEU: a method for automatic evaluation of machine translation*. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL '02). Association for Computational Linguistics, Stroudsburg, PA, USA, 311-318.

Ren, Z., LÜ, Y., Cao, J., Liu, Q., and Huang, Y. 2009. *Improving statistical machine translation using domain bilingual multiword expressions*. In Proceedings of the Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications (MWE '09). Association for Computational Linguistics, Stroudsburg, PA, USA, 47-54.

Sag, I., Baldwin, T., Bond, F., Copestake, A., and Flickinger, D. 2002. *Multiword Expressions: A Pain in the Neck for NLP*. In Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing (CICLing '02), Alexander F. Gelbukh (Ed.). Springer-Verlag, London, UK, UK, 1-15.

Schmid, H. 1994. *Probabilistic part–of–speech tagging using decision trees*. In Proceedings of International Conference on New Methods in Language Processing, Manchester, UK, 44–49.

Snover, M., Dorr, B., Schwartz, R., Makhoul, J., and Micciula, L. 2006. *A study of translation error rate with targeted human annotation*. In Proceedings of the Association for Machine Translation in the Americas Conference 2006, Boston, MA, USA, 223–231.

Zhang, Y., Vogel, S. 2010. *Significance Tests of Automatic Machine Translation Evaluation Metrics,* In Machine Translation: Volume 24, Issue 1 (2010), Page 51-65.