Sehda S^{2}MT: Incorporation of Syntax into Statistical Translation System

Yookyung Kim, Jun Huang, Youssef Billawala, Demitrios Master, Farzad Ehsani

Sehda Inc.
465 N. Fairchild Drive, Suite 123
Mountain View, CA 94043, USA
{kim,jun}@Sehda.com

Abstract

This paper describes Sehda's S^{2}MT (Syntactic Statistical Machine Translation) system submitted to the Korean-English track in the evaluation campaign of the IWSLT-05 workshop. The S^{2}MT is a phrase-based statistical system trained on linguistically processed parallel data.

1. Introduction

Sehda's S^{2}MT (Syntactic Statistical Machine Translation) system is a hybrid system which incorporates linguistic knowledge into statistical learning. The system learns phrase-to-phrase mapping and syntactic ordering separately. A feasibility test of the system is performed on the translation task presented by International Workshop on Spoken Language Translation (IWSLT) for Korean-to-English "Supplied+Tools" data track. We show that syntactic phrases are useful units to handle the structural problems of statistical Machine Translation and reduce the need for huge parallel text corpora.

2. Overview of S^{2}MT

Our system capitalizes on the intuition that language is broadly divided into two levels: structure and vocabulary. Structure is the syntax or relation among phrases that govern the formation of complex structures in a language. Vocabulary is the word-level representation of individual concepts in a language. In traditional approaches to Statistical Machine Translation (SMT), the system learns both types of information simultaneously. By separating the acquisition of structural information from the acquisition of vocabulary, however, an SMT system can learn both levels more easily and more efficiently. By modifying the existing corpus to isolate structure and vocabulary, we are able to take full advantage of all of the information content of the bilingual corpus, ultimately producing higher quality machine translation with less training data.

We separate the two levels of translation information by "chunking" [1] [2] the sentences in the bilingual corpus. Chunking is the process of separating the sentences into contiguous, structurally significant groups, such as noun phrases, verbal clusters, and prepositional phrases.[1] In contrast to full syntactic parsing employed in [5] [6], chunking is flexible enough to handle the ungrammaticalities of conversational data and provide us with syntactic information useful in handling structural issues [3].

Two learning passes are then performed[2]: one at the sentence level composed of phrase sequences to handle phrase reordering, and the other at the phrase level composed of word sequences to learn phrase translation properties. The results of the two learning passes are merged in the decoding step to produce translations, as shown in Figure 1.

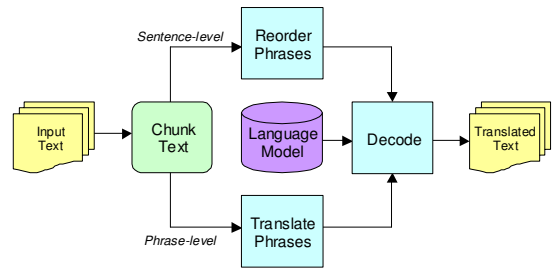

Figure 1: Overview of S^{2}MT

The system is composed of four modules:

(1) linguistic analyzer
(2) phrase translation module
(3) phrase reordering module
(4) decoding module which integrates phrases into a sentence

The linguistic analyzer consists of a parts-of-speech (POS) tagger, a morphological analyzer[3], a chunk parser, and a head word extractor. Except for a Korean morphological analyzer and Brill's English POS tagger, we use in-house tools. The head word extractor extracts the syntactic head word from a phrase to build a language model.

Phrase translation is done in two ways: (i) directly using phrase-to-phrase mapping; and (ii) statistically using IBM model-4 with training on the aligned phrases instead of aligned sentences. This is explained in greater detail below.

The significant word-order differences between Korean and English present a serious challenge to canonical SMT systems. In the S^{2}MT system, sentences are segmented into

[1] We use "chunks" and "phrases" interchangeably, unless otherwise noted.

[2] A similar approach is taken by [4] without much improvement on the baseline.

[3] We use Hangul Analysis Module (HAM) from Kukmin University for Korean morphological analysis. It separates a stem and suffixes in a word and tags them with a part of speech.

linguistically motivated phrases, which act as the fundamental units for reordering.

The system produces several candidate translations for each phrase, and they are decoded using linguistically augmented language models in conjunction with probabilities of different phrase orders, learned at the sentence level. Since each phrase is translated individually, without contextual information, it is vital to find a mechanism to communicate between phrases to find the best overall translation of the target sentence. We conducted a number of experiments using a variety of language modeling schemes, including the use of the lexical head of each phrase along with the syntactic type of the phrase.

2.1. Word Alignment

Word alignment based on parallel sentences plays an important role in SMT and acts as the first step in chunk alignment in Sehda's two-level approach. This alignment further generates a lexicon model necessary in subsequent processing.

We propose a learning algorithm to perform joint estimation of word alignment and lexicon model. In our approach, we first use GIZA++ to generate IBM model-based word alignments in both directions: Korean-to-English and English-to-Korean. We then construct an initial estimation of the probabilistic bi-lingual lexicon model based on the intersection or union of the GIZA++ word alignments. We use this lexicon model as the initial parameter set for our word re-alignment task. A maximum likelihood (ML) algorithm is further introduced using several different statistical source-target word translation models. The new word alignment is used as the source for the re-estimation of the lexicon model in the next iteration. We perform the joint estimation of the lexicon model and word alignment in an iterative fashion until a certain threshold criterion is reached.

In contrast to IBM models [7], our algorithm combines different lexicon model estimation approaches with the ML word alignment during each iteration of the model training. As a result, our system is more flexible in terms of the integration of the lexicon model and the word alignment during the recursive estimation, and thus can improve both predictability and precision of the estimated lexicon model and word alignment. Different probabilistic models are introduced in order to estimate the associativity between the source and target words. As a result, our approach is capable of increasing the recall ratio of word alignment and the lexicon size without decreasing the alignment precision, which is especially important for applications with limited training parallel corpus.

Given a source sentence $\vec{s} = s_1, s_2, \cdots, s_I$ and a target sentence $\vec{t} = t_1, t_2, \cdots, t_J$, we want to find the target word t_j which can be generated by source word s_i according to certain optimal criterion. Alignment between source and target words may be represented by an $I \times J$ alignment matrix $A = [a_{ij}]$, such that $a_{ij} = 1$ if s_i is aligned to t_j, and $a_{ij} = 0$ otherwise. The constrained ML based word alignment can be formulated as follows:

$$A^* = \arg\max_{A \in \Phi_L} p(\vec{t}, A \mid \vec{s}) \quad (1)$$

where Φ_L denotes the set of all possible alignment matrices subject to the lexical constraints.

The conditional probability of a target sentence generated by a source sentence depends on the lexicon translation model. We explored different ways to model the lexicon translation probability using the source-target word co-occurrence frequency, context information from the parallel sentence, and the alignment constraints. During each iteration of the word alignment, the lexical translation probabilities for each sentence pair are re-estimated using the lexical model learned from previous iterations, and the specific source-target word pairs occurring in the sentence. Furthermore, we introduced two types of lexical rules into word alignment. The first rule set consists of Korean case marking words, which should be aligned to the NULL word. The second rule set contains some incorrectly aligned bi-lingual lexicons generated from initial GIZA++ word alignment. Table 1 shows the comparison of word alignment performance between GIZA++ and Sehda's re-alignment tool. Realignment improves both precision and recall when both sentence context information and knowledge-based alignment constraints based on the union of GIZA++ initial alignments are used [8]. By combining the initial alignments with the context information and constraints, we achieve higher precision at the expense of only a slight decrease in recall. For chunk-based MT, the first realignment algorithm yields better translation results in terms of BLEU and NIST scores. This might be due to the fact that chunk coverage is more critical for our S^{2}MT system.

Table 1: Improvement of Word Alignment

	Precision	Recall
GIZA++	0.67	0.78
ReAlign (union+context+constraint)	0.73	0.87
ReAlign (intersection+context+constraint)	0.95	0.74

2.2. Chunk Alignment

To allow the two-level learning, both English and Korean sentences are segmented into syntactically meaningful phrases independently and the chunks are aligned. The resulting chunk alignment serves as the training data for statistical chunk translation as well as the direct phrase-to-phrase translation table. In the submitted system, we use a word alignment based method. If at least one word of a chunk in the source language is aligned to a word in a chunk in the target language by the improved word alignment using the two directions of GIZA++ training result, the two chunks are aligned.

A crucial difference between the S^{2}MT and other phrase based statistical MT model such as the Alignment Template

model [9] is that the source and target languages are independently chunk-parsed to find syntactically and semantically meaningful phrases in each language and then alignment is performed. In contrast to word alignment, chunk alignment tends to be one-to-one, and the non-monotonic alignments are still within manageable distance, as shown in Figure 2. In Figure 2, black represents word alignment and gray represent chunk alignment. As chunk alignment is guided by word alignment, gray areas include black squares. The chunk boundaries are defined in each language first, thus "go straight" is aligned to "곧바로가시" even though there is no word alignment between "straight" and "곧바로".

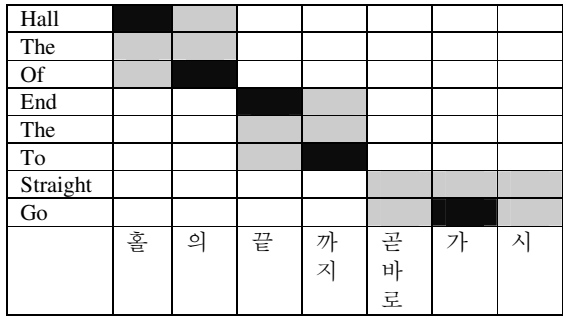

Figure 2: Example of Chunk Alignment

The chunking is performed using Sehda's rule-based chunk parser. The parser affords flexibility to accommodate idiosyncrasies of the language pair. For instance, Korean has a "missing argument" problem; pronouns are freely dropped as long as the reference can be resolved in context [10]. Missing pronouns must therefore be reintroduced when translating into English. By combining pronouns and verbal clusters in English into one chunk, the Korean verbal cluster with missing pronoun can be aligned to this chunk [4] as illustrated in Table 2.

Table 2: Example of Verbal Chunk Alignment

Korean chunk	English Chunk
싶/v +_/e	i/prp 'd/md like/vb
되/v +_/e	can/md i/prp
것/n 같/v +_/e	i/prp think/vbp

Combining pronouns and verbal clusters in English into one chunk can be harmful if an explicit pronoun appears in the Korean text. However, since verbal clusters in English will not be combined with non-pronoun NP subjects, there are simple verbal clusters available as translation candidates in addition to pronoun-verbal clusters. In more formal texts such as news paper articles where there are not many cases of dropped arguments, we find it is not necessary to combine subject pronouns and verbal clusters.

In the submitted system, we use only the improved word alignment from two directions of GIZA++ training, but there are many other possibilities to improve chunk alignment using dictionary and head-word alignments.

2.3. Phrase Translation

In our two-level leaning model, phrases are translated independently first, and then the best phrase is chosen among several candidate translations within context and phrases are reordered. In this section, we discuss the methods that we developed for phrase translation.

We apply two methods of phrase translation:
(1) **direct** phrase-to-phrase translation resulting from the chunk alignment,
(2) statistical translation using GIZA++ training on aligned chunks and **ReWrite** decoder.

The direct phrase translation uses the phrase translation model with probability constructed from the chunk alignment. The phrase translation probability is estimated by the co-occurrence frequency of the source-target chunk, and the unigram frequency of the source chunk from chunk alignment table. Direct phrase translation has the advantage of handling both word order within phrases as well as translations of non-compositional expressions, which covers many translation divergences [11]. While the quality of direct phrase translation is very high, the coverage may be low, as it depends on the size and overlap of the training corpus. Several ways of chunking with different rules are tested to construct a better direct phrase translation table to balance quality and coverage. For the IWSLT development set, we achieved a coverage of 72%. The coverage is defined as the ratio of non-punctuation chunks which are translatable by direct translation to the total non-punctuation chunks in the development set.

The second method (ReWrite) [12] makes use of the pre-existing decoder ReWrite and IBM model-4. Though the model and decoding program have already been developed by IBM, our training material is different from IBM's. The system learns word translation probability from aligned chunk phrases instead of entire aligned sentences. Since chunk phrases on average consist of 2.5 words,[5] the complexity to learn word translation probabilities will be reduced significantly. This is important because the traditional translation statistics must consider the probability of every word in the input sentence mapping onto every word in the output sentence. In common training data it is not unusual to see sentences of 30 or more words in the written texts, so this is a significant number of probabilities to consider. If we can perform the word-level training on aligned chunks instead of aligned sentences, then the simplification of the translation model will be significant. Hence more accurate mapping is expected without an increase in size of the bilingual corpus. In addition, the translation is to produce a chunk rather than the entire sentence at this step, thus a better distortion model is learned when we train using aligned chunks instead of the aligned sentences.

[4] The missing argument problem is pervasive in the training and development set because they are conversational data, but there were unexpected many pronouns in the test set. We are not sure whether the pronouns in the test set are artificially added or not.

[5] The number of words in a chunk depends on the exact definition of chunk used.

One consideration when using aligned chunks instead of aligned sentences is that we may lose some training data due to incorrect chunk alignment. This is particularly true for small training corpora, as is the case with the IWSLT evaluation. To overcome this problem, both the aligned chunks and aligned sentences are used, but only the English chunks are used in language modeling for phrase translation.

In general, the translation quality of phrases by this purely statistical model (ReWrite) is worse than the direct phrase translation, as illustrated in Table 3, though coverage is close to 100% except for instances of unknown words.

Table 3: Example of Direct and ReWrite Translation

	Direct Translation	ReWrite Translation
하룻밤/n+에/j	per/in night/nn	at/in night/nn
하룻밤/n +에/j	a/dt night/nn	in/in night/nn
하룻밤/n +에/j		on/in night/nn
하룻밤/n +에/j		in/in a/dt night/nn
창가/n 쪽/n	a/dt table/nn near/in	a/dt window/nn
자리/n +로/j	the/dt window/nn	seat/nn
창가/n 쪽/n	a/dt window/nn	window/nn side/nn
자리/n +로/j	seat/nn	seat/nn
추천/n + 하/t + 시/f + _/e	you/prp recommend/vb	thank/vb you/prp recommend/vb
		highly/rb thank/vb you/prp
		your/prp$ you/prp recommend/vb

2.4. Reordering of Phrases

Word-based SMT works poorly for language pairs as Korean and English which are structurally very different, as the distortion models are capable of handling only local movement of words. The proposed model's unit of reordering is the syntactic phrase. The performance of reordering in our model is superior to word-based SMT both in quality and speed due to the reduction in search space. To evaluate the reordering per se, we first used the ideal translation of phrases that are found from reference translations. Table 4, "before" indicates the English phrases in Korean word order, and "after" the result of reordering, which is in English word order.

Table 4: Example of Reordering

before	a menu	please	show me
after	show me	a menu	Please

We model the phrase reordering problem as the combination of traveling salesman problem (TSP) and global search of the ordering of the target language phrases. The TSP problem is an optimization problem which tries to find the path to cover all the nodes in a direct graph with certain defined cost function. For phrase re-ordering of machine translation, we use the language model (LM) score between contiguous chunks as the transitional cost between two phrases. Our LM score is obtained through the log-linear interpolation of an n-gram based lexicon LM, and an n-gram based phrase chunk head LM. We use a 3-gram LM with Good-Turing discounting to train the target language LM. The LM training data is the English part of IWSLT05's training set which contains 20k sentences from the BTEC corpus. We also added POS tags to the 20k training sentences and the test sentences, in order to compensate for the lexical coercion[6] phenomenon in machine translation.

Due to the efficiency of our combined global search and TSP algorithm, we didn't use a distortion model to guide the search for optimal phrase reordering paths. Reordering results are shown in Table 5. The system before reordering is simply combining top-1 phrase translations of each source phrase without considering context, and the system after reordering is the result after statistical reordering is performed, without considering n-best translations of each phrase.

Table 5: Experimental results of chunk reordering

	NIST score	BLEU score
before	5.7603	0.1641
after	6.1290	0.2147

2.5. Decoding

Sehda's MT decoder, as depicted in Figure 3, is a chunk-based hybrid decoder. During the decoding stage, N-best chunk translation candidates from both direct table (DT) and ReWrite (RW) tables are produced from the phrase translation module. The associated probabilities of these translated chunks are first normalized to the global distributions of DT and RW chunks separately and subsequently merged using optimized contribution weights.

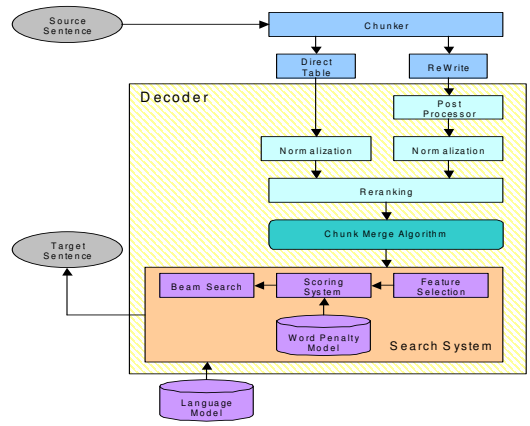

Figure 3: Decoder System Diagram

A surface-form language model trained on the corpus is used to predict the probability of any sequence of DT and RW chunks. A word penalty model is necessary to compensate for the fact that the LM systematically penalizes longer target chunks.

[6] Lexical coercion is a phenomenon in natural language processing that we condition translation of a foreign word on the source word and its parts-of-speech.

3. Experimental Results

In this section, we present results on IWSLT's Korean-English task with limited training corpus. The training data was provided by the organizer and consists of 20k parallel sentences from BTEC corpus. The development set consists of 506 Korean sentences with 16 references, and the test set consists of 506 Korean sentences from the same domain as the training data. We note that the Korean test set has an unconventional tokenization different from the training set. For the proper morphological analysis and chunking, we modified the tokenization consistent with the training set. Table 6 summarizes some statistics of the training/test data.

Table 6: Statistics of training/development/test sets

Corpus		Korean	English
Training Set	Vocabulary size	8.9k	8.7k
	# of learned chunk pairs	32.6k	32.6k
Development Set	# of chunks	2367	2367
	DT coverage	72%	
Test Set	# of chunks	2422	2422
	DT coverage	71%	

To assess the contribution of linguistic processing, the performance of IBM model-4 with no text processing is compared to those with processing. The results shown in Table 7 are based on the development set.

Table 7: Systems Comparison on the development set

	SMT-plain	SMT-pos	SMT-trans	S^2MT
NIST	2.049	4.906	5.294	6.3412
BLEU	0.134	0.173	0.190	0.2491

SMT-plain is the result of IBM model-4 trained on the given training set. SMT-pos is the result of the same model on the modified training set with parts-of-speech tagging on the English side and parts-of-speech tagging and stemming on the Korean side. The big improvement of performance from SMT-plain to SMT-pos is due to the fact that Korean is a morphologically rich language and morphological processing such as detaching suffixes and stemming reduces the number of lexical items and hence results in a better lexical translation model. For a language pair with less complexity in morphology, we do not expect such a big performance improvement with simple addition of linguistic processing.

SMT-trans is a further modification of the training set with heuristic transformations [13] on the Korean side. Chunking and heuristic reordering are performed: the direct object phrases are moved after the verb and the auxiliary verbs are moved before the main verb. The S^2MT includes all the linguistic processing plus statistical reordering and decoding. The contribution of statistical reordering and decoding was significant as shown in Table 8.

Table 8: Contribution of Each Component of S^2MT

	S^2MT top 1	S^2MT - reordering	S^2MT - Decoding
NIST	5.7603	6.1290	6.3412
BLEU	0.1641	0.2147	0.2491

Before these two steps, we simply chose the best translation for each phrase without considering the context and measured the translation quality (S^2MT top-1). S^2MT-reordering is done by reordering the context independent best phrase translations according to a language model. After an optimal ordering is chosen,all n-best chunk translation candidates are considered in S^2MT-decoding. To see the contribution of each module we separate reordering and decoding, however, a fully integrated system should achieve superior results.

The best system for the development set is used for the test set evaluation and the results are shown in Table 9, where BLEU [14], NIST [15], GTM [16], METEOR [17], WER, and PER [18] are 6 automatic evaluation metrics used in the evaluation campaign.

Table 9: Test Set Result

	S^2MT
NIST	6.511
BLEU	0.2064
WER	0.7033
PER	0.5470
METEOR	0.5111
GTM	0.4224

4. Conclusions

In this paper, we present Sehda's S^2MT system which incorporates linguistic knowledge into statistical machine translation. We show that we can perform translation with reasonable quality using very limited resources. From our experiments, linguistic processors such as a morphological analyzer and a chunk parser significantly reduce the dependence on training parallel corpus. Our new word alignment algorithm can improve both precision and recall through the incorporation of lexical knowledge and the interaction between the lexicon model and word alignment during the learning stage. The improved word alignment, together with new chunking rules, help us obtain an improved NIST score in IWSLT supplied data track evaluation. The combined optimal search and TSP solution helps us solve the phrase reordering problem in a timely fashion without significant degradation of the performance.

The S^2MT system was developed for a NSF SBIR project in news text translation domain. We made minor adaptations to the system for our participation of IWSLT evaluation. It remains a future research task to use a phrase based SMT [19] [20] instead of a word based SMT as a baseline and to find out the additional value of syntactic information.

5. Acknowledgements

This research was funded by NSF SBIR award #0441891. We also want to thank NIST Advanced Technology Program (ATP) for the support of the decoder part of S^2MT system. We further like to thank the reviewers for their helpful comments.

6. References

[1] Abney, S. P., "Parsing by Chunks." In Robert C. et al eds., *Principle-Based Parsing: Computation and Psycholinguistics*, pages 257--278. 1991

[2] Sang, T. K. and Buchholz, S., "Introduction to the CoNLL 2000 Shared Task: Chunking", in *Proceedings of CoNLL-2000*, pp. 151 – 153, Lisbon, Portugal 2000.

[3] Watanabe, T., Sumita, E., and Okuno, H.G, "Chunk-based Statistical Translation," in *Proceedings. of the 41st Annual Meeting of the Assoc. for Computational Linguistics*, pp. 303-310. 2003.

[4] Scafer, C., and Yarowsky, D., "Statistical Machine Translation Using Coercive Two-Level Syntactic Transduction," in *Proceedings of ACL*, pp. 9-16. 2003.

[5] Yamada, K. and Knight, K., "A Syntax-Based Statistical Translation Model," ACAL, 2001.

[6] Och, F. et al. "Syntax for Statistical Machine Translation", Final Report of John Hopkins 2003 summer workshop".

[7] Brown, P. F., Della Pietra, S.A., Della Pietra, V.J., and Mercer, R. L., "The Mathematics of Statistical Machine Translation: Parameter Estimation," *Computational Linguistics*, Vol. 2, Issue 9, pp. 263 – 311, 1993.

[8] Lee, S. and Roukos, S. "IBM Spoken Translation System Evaluation," IWSLT-2004. 2004.

[9] Och, F.J., Tillmann, C., and Ney, H., ""Improved Alignment Models for Statistical Machine Translation," in *Proc. of the Joint SIGDAT Conf. On Empirical Methods in Natural Language Processing and Very Large Corpora*, University of Maryland, College Park, MD, pp. 20 – 28, 1999.

[10] Han, C., Levoie, B., Palmer, M., Fambow, O., Kittredge, R., Korelskly, T., and Kim, M., "Handling Structural Divergences and Recovering Dropped Arguments in a Korean/English Machine Translation System." 1999.

[11] Habash, N. and Dorr, B., "Handling Translation Divergences: Combining Statistical and Symbolic Techniques in Generation-Heavy Machine Translation," In *Proc. Fifth Conference of the Association for Machine Translation in the Americas*, Tiburton, CA 2002.

[12] Germann, U., "Greedy Decoding for Statistical Machine Translation in Almost Linear Time," in *Proceedings of HLT-NAACL*, Edmonton, AB, Canada, 2003.

[13] Matuson E., Popovic M., Zens R., and Ney H., "Statistical Machine Translation of Spontaneous Speech with Scarce Resources," in *Proceedings of IWSLT*, Kyoto, Japan. 2004.

[14] Papineni, K., Roukos, S., Ward, T., and Zhu, W., "BLEU: A Method for Automatic Evaluation of Machine Translation," Technical Report RC22176 (W0109-022), IBM Research, 2001.

[15] Doddington, G. "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," In Human Language Technology: Notebook, Proceedings, pp. 128-132. San Diego, CA2002.

[16] Melamed, D., Green, R., and Turian, J., "Precision and Recall of Machine Translation," in *Proceedings of HLT-NAACL*, 2004.

[17] Banerjee, S., and Lavie, A., "METEOR: An Automatic Metric for MT Evaluation with improved correlation with Human Judgments," in *Proceedings of Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization, 43th Annual Meeting of ACL*, Ann Arbor, Michigan, June 2005.

[18] Och, F., *Statistical Machine Translation: From Single Word Models to Alignment Templates*. Ph.D. thesis, RWTH Aachen, Germany, 2002.

[19] Chiang, D., "A Hierachical Phrase-Based Model for Statistical Machine Translation," ACL pp.263-270. 2005.

[20] Zhang, Y., Vogel, S. and Waibel, A., "Integrated Phrase Segmentation And alignment Algorithm for Statistical Machine Translation," in *Proceedings of International Conference on Natural Language Processing and Knowledge Engineering*.2003.