The RWTH Aachen Machine Translation system for IWSLT 2010

Saab Mansour, Stephan Peitz, David Vilar, Joern Wuebker and Hermann Ney

Human Language Technology and Pattern Recognition
Computer Science Department
RWTH Aachen University
Aachen, Germany

surname@cs.rwth-aachen.de

Abstract

In this paper we describe the statistical machine translation system of the RWTH Aachen University developed for the translation task of the IWSLT 2010. This year, we participated in the BTEC translation task for the Arabic to English language direction. We experimented with two state-of-the-art decoders: phrase-based and hierarchical-based decoders. Extensions to the decoders included phrase training (as opposed to heuristic phrase extraction) for the phrase-based decoder, and soft syntactic features for the hierarchical decoder. Additionally, we experimented with various rule-based and statistical-based segmenters for Arabic.

Due to the different decoders and the different methodologies that we apply for segmentation, we expect that there will be complimentary variation in the results achieved by each system. The next step would be to exploit these variations and achieve better results by combining the systems. We try different strategies for system combination and report significant improvements over the best single system.

1. Introduction

This paper describes the statistical machine translation (SMT) system used for our participation in the 2010 International Workshop on Spoken Language Translation (IWSLT 2010). We used it as an opportunity to incorporate novel methods which have been investigated at RWTH over the last year and which have proven to be successful in other evaluations. We participated in the Arabic-English BTEC task, and used standard alignment and training tools as well as our in-house phrase-based and open-source hierarchical SMT decoders.

We explored and implemented different segmentation tools for Arabic. The methods used to implement those tools vary from rule-based methods (typically encoded as finite state transducers) such as [1], to methods which are statistically-based such as [2] and [3]. All these works have shown that segmentation improves MT quality significantly for both small and large scale tasks.

Due to the different methodologies that we apply for segmentation, we expect that there will be complimentary variation in the results achieved by each method. The next step would be to exploit those variations and achieve better results by combining the systems.

This paper is organized as follows. In Section 2, we present the data and resources that will be used to build our segmenters and the SMT system. In Section 3, we discuss the problems of Arabic SMT and present the solution of segmentation including the different methods applied in this work. The phrase-based system and the hierarchical system including extensions will be described in Section 4 and Section 5 respectively. Evaluation and discussion of the results of the various segmentation methods will be presented in Section 6. In Section 7, we briefly introduce the system combination framework used in this work. A discussion of the results and further examples including final remarks are given in Section 8.

2. Experimental setup

2.1. Arabic word segmentation

To train the segmentation methods, we use the Arabic Treebank Part 1 v3.0[1]. The treebank contains 150 000 word tokens and is drawn from the news genre. The Arabic words are segmented according to the so-called ATB scheme. In this scheme, prepositions (excluding the Arabic determiner *Al* and the future marker *s* [2]) and possessive and objective pronouns are split from the Arabic stem.

For some models, we use a lexicon to limit the choice of possible segmentations. For this purpose, we use the Buckwalter Arabic Morphological Analyzer (BAMA) v1.0[3], a rule based analyzer, with 80 000 lexicon entries.

2.2. MT data

For training the SMT systems, we use the official IWSLT 2010 training data augmented with the IWSLT03 and

[1] LDC Catalog No. LDC2005T02

[2] Arabic characters are encoded using the Buckwalter transliteration: http://www.qamus.org/transliteration.htm

[3] LDC Catalog No. LDC2002L49

IWSLT07 test sets. We only use the two longest references of the test sets, as this proved to achieve the best MT quality on initial experiments.

English preprocessing includes tokenization and casing the first word of the sentence according to the most frequent form in the training data (frequent casing). Arabic preprocessing includes removal of short vowels and tokenization.

3. Arabic segmentation

Written Modern Standard Arabic (henceforth *Arabic*) is known for its complex morphology and ambiguous writing system. These complexities are expressed in an SMT system at several levels. The first step in most of state-of-the-art SMT systems, after processing the bilingual corpora, is to generate an alignment between the source and the corresponding target (translation) sentence. Form these alignments a word lexicon and more importantly a phrase lexicon (usually using heuristics) are extracted. In Arabic, one word often corresponds to more than one word in traditional target languages such as English and French, posing a problem to the traditional IBM alignment models. Those complex Arabic words are generated from the attachment of a stem to prefix, affix and suffix clitics. Segmenting a word into its corresponding morphemes is already an ambiguous process and relies not only on grammatical rules, but also on the context of the word at hand. Ambiguity is even a harder problem in Arabic, expressed in the lack of short vowels in written Arabic and the high-degree of grammatical inflection. The increase of ambiguity is expressed in the increased number of possible translations per word, but, in addition, it is expressed in the possible segmentations of the word which eventually affects the corresponding translations.

A well studied solution of the problems mentioned above is Arabic word segmentation. Splitting an Arabic word into its corresponding prefixes, stem and suffixes lessens the number of out-of-vocabulary (OOV) words, resolves some of the ambiguous Arabic words and generates more one-to-one correspondences between the Arabic side and the target language side which can be easily captured by the IBM alignment models.

In this work, we experimented with the following segmenters:

- FST - A Finite State Transducer-based approach introduced and implemented by [1]. The FST is used as a framework to implement a set of rules for segmentation of Arabic. The prefixes that are split include w,f,k,l,b,Al and s. Suffixes which are segmented are pronouns (objective and possessive). The method is characterized by fast processing speed but suffers from the lack of context in the decision procedure leading to erroneous output.
- SVM - we reimplemented the classifier suggested by [4]. In their method, each character is classified by its segment rule (prefix, stem and suffix) and position (beginning and inside segment). Arabic words are segmented according to the ATB scheme. Additionally, feminine marker normalization (tX→p+X) using an SVM model is applied on top of the segmenter output, which proved to be significant for the performance of MT in our experiments.
- CRF - we implemented a CRF classifier for segmentation using similar setup of classifiers and classes as in the SVM model. The software we use as an implementation of conditional random fields is named CRF++[4].
- MorphTagger - is a general architecture for Part-Of-Speech (POS) tagging of natural languages. The architecture was first proposed in [5] and applied for the task of POS tagging of Hebrew. [6] adapted the architecture to the Arabic language. MorphTagger is implemented using Buckwalter Arabic Morphological Analyzer v1.0 (BAMA) as a morphological analyzer and a Hidden-Markov-Model (HMM) (using the SRIML[5] toolkit) as the disambiguator component.
- MADA - The Morphological Analysis and Disambiguation of Arabic (MADA) system, developed in [7], can be seen as an extension of an SVM-based system with the incorporation of a morphological analyzer. As in [8], we experiment with different segmentation schemes for each chosen analysis. We use the schemes directly implemented in the MADA version we are using, namely: D1,D2,D3 and the ATB (TB) schemes.

4. Phrase-based system

4.1. Standard phrase-based system (PBT)

The phrase-based SMT system used in this work is an in-house implementation of state-of-the-art phrase-based MT system as described in [9]. We use the standard set of models with phrase translation probabilities for source-to-target and target-to-source direction, smoothing with lexical weights, a word and phrase penalty, distance-based and lexicalized reordering and an n-gram target language model.

4.2. Phrase training (Forced Alignment-FA)

To estimate the phrase translation probabilities we experimented with both standard heuristic phrase extraction ([10]) and a forced alignment training procedure as described in [11]. The latter estimates the probabilities as relative frequencies from the phrase-aligned training data, which is computed by a modified version of the translation decoder. To do this, the translation decoder is constrained to produce the reference translation for each bilingual sentence pair. In order to counteract overfitting, leaving-one-out is applied in training. In addition to providing a statistically well-founded

[4] http://crfpp.sourceforge.net/

[5] http://www-speech.sri.com/projects/srilm/

Table 1: AR-EN BTEC 2010: IWSLT08 results summary (nocase+punc)

	PBT		FA		JANE		SYN		POMS	
System	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER
CRF	55.5	29.8−	56.4−	30.7	55.4	30.6	55.7	30.8	56.1	30.2
FST	54.5	30.7	55.9	30.3	55.3	30.2	54.4	31.2	56.0−	29.4−
MADA ATB	55.1	29.5	57.1+	29.2+	55.2	29.4	55.7	29.4	55.2	29.9
MADA D1	54.8	30.8	55.2−	30.6−	53.9	31.2	54.5	30.9	54.8	30.8
MADA D2	55.4	29.9	55.5	30.1	54.6	30.2	54.8	31.2	55.5−	29.7−
MADA D3	55.4	29.6	56.5	30.1	56.6	28.8|	56.5	28.5+	56.8−	28.7
MorphTagger	56.5|	29.2|	55.8	30.1	57.1|	29.4	56.6|	29.2	57.5∗	28.5∗
SVM	56.1	29.7	55.9	30.0	56.6−	28.9−	55.4	30.3	54.9	29.5
TOK	55.5−	30.1−	54.8	30.3	53.0	32.4	52.7	32.5	53.4	32.3

phrase model, the forced alignment procedure has the benefit of producing smaller phrase tables.

5. Hierarchical system

5.1. Standard hierarchical system (JANE)

We used the open source hierarchical phrase-based system Jane, developed at RWTH and free for non-commercial use [12]. This approach is an extension of the phrase-based approach, where the phrases are allowed to have gaps [13]. In this way long-range dependencies and reorderings can be modelled in a consistent statistical framework.

The system labelled as JANE represents a fairly standard setup of the system and constitutes a baseline upon which the two next systems are built.

5.2. Soft syntax labels (SYN)

To extend the hierarchical system with syntax information of the English target side, we derive soft syntactic labels as in [14] with the modifications described in [15]. In this model, instead of considering only a single, generic non-terminal in the underlying grammar, we extend the set of labels to include syntactic categories as found in syntactic parse trees. To extract the syntax information, we parse the English target sentences with the Stanford parser[6].

It is important to note that the new non-terminals are considered in a probabilistic way. In this way, the parsing process itself continues to use the generic non-terminal as in the baseline model and the parsing space is unaltered. The extended set of non-terminals is then used to compute a new probabilistic feature that measures the well-formedness of the translation with respect to the syntactic constructs.

5.3. Poor-man syntax (POMS)

In this approach we apply the same model as described in the previous section, but the method for producing the new non-terminals is altered as described in [16]. Instead of relying on parse trees based on linguistic knowledge we rely on automatic clustering methods. This makes this approach applicable also for underresourced languages for which no linguistic tools may be available.

6. Results

The results of the different segmentation methods and schemes are summarized in Table 1. In this table, the best result in a column is marked with |, thus comparing different segmentations for the same decoder setup. We mark with − the best (in a row) performing decoder over a specific segmentation method. ∗ marks the best result overall in the table. For comparison purposes to the proposed segmenters, we include a TOK "segmenter" for Arabic which performs punctuation tokenization only. In our experiments, we use IWSLT04 for development (automatic tuning of the translation system weights) and IWSLT08 for comparison between the systems. We include both BLEU and TER to measure the MT systems translation quality.

From the raw results, we observe that segmentation usually helps. In the case of the FSA method, the inconsistent segmentations are causing a high rate of OOV words therefore inferior results. The MADA D1 scheme is characterized by very low degree of segmentation (only the conjunction clitics *f* and *w* are split) which proves insufficient for the small task at hand.

Comparing the different decoders setups, we notice that the hierarchical decoders have the upper hand in most of the cases, namely FST, MADA D2 and D3, MorphTagger and SVM, while the phrase-based decoders are performing better for the CRF, TOK and MADA ATB and D1 segmentations. For the segmentation methods, we observe that MorphTagger is performing better in most of the measures, namely PBT, JANE, SYN BLEU and POMS. MADA D3 is performing best for JANE TER and SYN TER, and MADA ATB is best for the FA enhanced decoder.

[6] http://nlp.stanford.edu/software/lex-parser.shtml

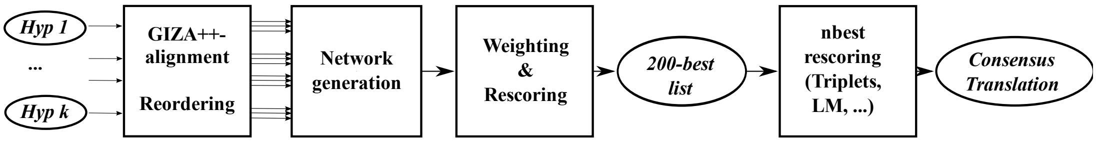

Figure 1: The system combination architecture.

7. System combination

The pipeline of the system combination is based on the pipeline described in [17], which was used in the WMT 2010 evaluation and achieved state-of-the-art results.

Figure 1 gives an overview of the system combination architecture. After preprocessing the MT hypotheses, pairwise alignments (using GIZA++) between the hypotheses are calculated. The hypotheses are then reordered to match the word order of a selected primary or skeleton hypothesis. From this, a lattice is created which is then rescored using system prior weights and a language model. We use the IWSLT05 test set to tune the weights for lattice rescoring. The single best path in this confusion network then constitutes the consensus translation which is outputted by this system. The consensus translation is then true cased and postprocessed.

We were interested in three kinds of combinations: *(i)* different segmentations of the same decoder setup; *(ii)* different decoder setups of the same segmentation method; and *(iii)* mixture of segmentations and decoder setups combination.

The results of combining different segmentations with the same decoder setup are summarized in Table 2. In this table, we include the best single system on IWSLT08 for each decoder. We compared two combination strategies. Similar to [8], we combined the different schemes of the MADA segmenter. In this case, we achieve improvements of up to +1.2% BLEU and -1.1% TER over the best single system. Next, we tried the combination of the outputs of the different segmentation "methods". This gave an improvement of up to +1.9% BLEU and -1.8% TER. Furthermore, we tried a combination of all the schemes and methods ("Schemes+Methods"), but here the results were mixed and no clear conclusion could be drawn. For comparison purposes, we took the best performing segmentation for each decoder setup, and combined these together (Best systems combi.). This did not result in further improvements.

In Table 3, we summarize the results of combining different decoder setups for each segmentation. This kind of combination resulted in improvements of up to +1.3% BLEU and -1.2% TER (except for TOK and MADA ATB cases). We also took the best performing decoder per segmentation and combined those together (Best segmentations combi.). In contrast to "Best systems combi.", an improvement was observed, probably due to the fact we combine more systems (7 for the "Best segmentations combi." versus 5 for each "Combi SEG").

Table 2: AR-EN BTEC 2010: Segmentations combination per decoder-setup results (nocase+punc). For each decoder, the best segmentation, a combination of the MADA schemes, a combination of the segmentation methods, and a combination of both are displayed.

	IWSLT08	
System	BLEU	TER
Best PBT (MorphTagger)	56.5	29.2
PBT MADA Schemes	57.4	28.4
PBT Methods	58.0	28.3
PBT Schemes+Methods	58.3	28.0
Best FA (MADA ATB)	57.1	29.2
FA MADA Schemes	57.8	28.8
FA Methods	58.1	28.7
FA Schemes+Methods	58.7	28.3
Best JANE (MorphTagger)	57.1	29.4
JANE MADA Schemes	58.0	28.3
JANE Methods	59.0	27.8
JANE Schemes+Methods	57.3	28.4
Best SYN (MorphTagger)	56.6	29.2
SYN MADA Schemes	57.8	28.1
SYN Methods	57.7	28.2
SYN Schemes+Methods	57.4	28.4
Best POMS (MorphTagger)	57.5	28.5
POMS MADA Schemes	57.3	28.1
POMS Methods	59.2	27.4
POMS Schemes+Methods	58.8	27.7
Best systems combi.	58.2	27.8

Last, we performed a combination of different decoders and segmentations. We combined the best n systems, for $5 \leq n \leq 20$. The best result was achieved for $n = 15$, scoring 59.8% BLEU and 27.1% TER. Then, we studied the effect of removing each of the systems from the combination. The system that removing it gave the best overall gain was suppressed. We repeated this process until no further improvement was achieved. The systems that entered the final mixture were: PBT MorphTagger, FA CRF, FA MADA D3, FA MADA ATB, JANE MADA D3, JANE SVM, SYN MADA D3, SYN MorphTagger, POMS MorphTagger, and POMS MADA D3. This system is reported in Table 4, and was submitted as our primary submission for this year evaluation.

Table 3: AR-EN BTEC 2010: Decoders combination per segmentation results (nocase+punc). For each segmentation, the best decoder and a combination of the decoders are displayed.

	IWSLT08	
System	BLEU	TER
Best CRF (FA)	56.4	30.7
Combi CRF	57.3	28.6
Best FST (POMS)	56.0	29.4
Combi FST	56.8	29.3
Best MADA ATB (FA)	57.1	29.2
Combi MADA ATB	56.9	28.5
Best MADA D1 (FA)	55.2	30.6
Combi MADA D1	55.2	30.2
Best MADA D2 (POMS)	55.5	29.7
Combi MADA D2	56.7	29.0
Best MADA D3 (POMS)	56.8	28.7
Combi MADA D3	58.1	27.5
Best MorphTagger (POMS)	57.5	28.5
Combi MorphTagger	58.1	28.2
Best SVM (JANE)	56.6	28.9
Combi SVM	57.3	29.4
Best TOK (PBT)	55.5	30.1
Combi TOK	54.8	31.1
Best segmentations combi.	59.4	27.4

Table 4: AR-EN BTEC 2010: Submitted system. IWSLT08 is reported with nocase+punc, IWSLT09 and IWSLT10 are the official case+punc results.

	Submitted system	
System	BLEU	TER
IWSLT08	60.2	26.7
IWSLT09	55.3	27.7
IWSLT10	46.6	32.7

8. Conclusions and outlook

In our participation in the IWSLT 2010 evaluation, we compared several publicly available Arabic segmentation methods and translation decoders setups for the task of SMT. Supporting the outcome of previous work, we found that high-degree of segmentation performs better than simple tokenization on a small scale Arabic to English translation task. Nevertheless, the differences between the high-degree segmentation methods proved to be statistically insignificant.

Next, we experimented with exploiting the advantages of the different segmentation-based SMT systems through system combination. We start out by combining several segmentation schemes of the same model. By this strategy, we achieve improvements over the best single system. Next, we tried a different strategy, where we combined the different segmentation methods rather than different segmentation schemes. In this case, we obtained better results over the schemes combination method. Finally, a mixture of decoders and segmentation schemes and methods had another improvement and the best result overall.

9. Acknowledgements

This work was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation and partly based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-08-C-0110.

10. References

[1] A. El Isbihani, S. Khadivi, O. Bender, and H. Ney, “Morpho-syntactic Arabic Preprocessing for Arabic to English Statistical Machine Translation,” in *Proceedings on the Workshop on Statistical Machine Translation*. New York City: Association for Computational Linguistics, June 2006, pp. 15–22.

[2] Y.-S. Lee, “Morphological analysis for statistical machine translation,” in *HLT-NAACL ’04: Proceedings of HLT-NAACL 2004*. Morristown, NJ, USA: Association for Computational Linguistics, 2004, pp. 57–60.

[3] N. Habash and F. Sadat, “Arabic preprocessing schemes for statistical machine translation,” in *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*. New York City, USA: Association for Computational Linguistics, June 2006, pp. 49–52.

[4] M. Diab, K. Hacioglu, and D. Jurafsky, “Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks,” in *HLT-NAACL 2004: Short Papers*. Boston, Massachusetts, USA: Association for Computational Linguistics, May 2 - May 7 2004, pp. 149–152.

[5] R. Bar-haim, K. Sima'an, and Y. Winter, "Part-of-speech tagging of modern Hebrew text," *Nat. Lang. Eng.*, vol. 14, no. 2, pp. 223–251, 2008.

[6] S. Manour, K. Sima'an, and Y. Winter, "Smoothing a lexicon-based POS tagger for Arabic and Hebrew," in *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 97–103.

[7] N. Habash and O. Rambow, "Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop," in *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2005, pp. 573–580.

[8] F. Sadat and N. Habash, "Combination of Preprocessing Schemes for Statistical MT," in *Proc. of the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*, Sydney, Australia, July 2006, pp. 1–8.

[9] R. Zens and H. Ney, "Improvements in Dynamic Programming Beam Search for Phrase-based Statistical Machine Translation," in *International Workshop on Spoken Language Translation*, Honolulu, Hawaii, Oct. 2008, pp. 195–205.

[10] F. J. Och, C. Tillmann, and H. Ney, "Improved Alignment Models for Statistical Machine Translation," in *In Proc. of the Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP99)*, College Park, MD, June 1999, pp. 20–28.

[11] J. Wuebker, A. Mauser, and H. Ney, "Training phrase translation models with leaving-one-out," in *Proceedings of the 48th Annual Meeting of the Assoc. for Computational Linguistics*, Uppsala, Sweden, July 2010, pp. 475–484.

[12] D. Vilar, D. Stein, M. Huck, and H. Ney, "Jane: Open source hierarchical translation, extended with reordering and lexicon models," in *ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR*, Uppsala, Sweden, July 2010, pp. 262–270.

[13] D. Chiang, "Hierarchical phrase-based translation," *Computational Linguistics*, vol. 33, no. 2, pp. 201–228, June 2007.

[14] A. Venugopal and A. Zollmann, "Preference Grammars: Softening Syntactic Constraints to Improve Statistical Machine Translation," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Boulder, Colorado, USA, June 2009, pp. 236–244.

[15] D. Stein, S. Peitz, D. Vilar, and H. Ney, "A cocktail of deep syntactic features for hierarchical machine translation," in *Conference of the Association for Machine Translation in the Americas 2010*, no. 9, Oct. 2010, accepted for publication.

[16] D. Vilar, D. Stein, S. Peitz, and H. Ney, "If I only had a parser: Poor man's syntax for hierarchical machine translation," in *International Workshop on Spoken Language Translation*, Paris, France, Dec. 2010, accepted for publication.

[17] G. Leusch and H. Ney, "The RWTH system combination system for WMT 2010," in *ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR*, Uppsala, Sweden, July 2010.