The DCU Machine Translation Systems for IWSLT 2010

Hala Almaghout, Jie Jiang, Andy Way

CNGL, School of Computing
Dublin City University
Dublin, Ireland
{halmaghout, jjiang, away}@computing.dcu.ie

Abstract

In this paper, we give a description of the DCU machine translation systems submitted to the evaluation campaign of The International Workshop on Spoken Language Translation (IWSLT) 2010. We participated in the BTEC Arabic-to-English task in addition to the DIALOG task for translation between English and Chinese in both directions. We explore different extensions to Phrase-Based and Hierarchical Phrase-Based Machine Translation Systems. We deploy a paraphrase system as an extension to our English-to-Chinese Phrase-Based translation system. For the Hierarchical Phrase-Based system, two different syntactic augmentation methods are investigated: the first is Syntax Augmented Machine Translation ,which is based on constituent grammar, while the other one is based on Combinatory Categorial Grammar. In addition, we combine the output of our hierarchical systems using a system combination method based on confusion networks.

1. Introduction

In this paper we describe the machine translation systems built for our participation at IWSLT 2010. We investigate and compare several extensions to Phrase-Based [1] and Hierarchical Phrase-Based (HPB) [2] Statistical Machine Translation Systems.

A paraphrase system is explored as an extension to the Phrase-Based system. We conduct experiments comparing its performance with a Phrase-Based baseline system on English-to-Chinese translation. For the Hierarchical Phrase-Based system, we try two syntax augmentation methods. The first one is Syntax Augmented Machine Translation (SAMT) [3], which uses constituent grammar to attach syntactic labels to nonterminals in hierarchical rules. The other method uses Combinatory Categorial Grammar (CCG) instead of constituent grammar for nonterminal labeling. We compare these two methods with the Hierarchical Phrase-Based System baseline for Arabic-English and Chinese-English translation. Furthermore, we try to combine the outputs of the two syntax augmented Hierarchical Phrase-Based systems with the baseline Hierarchical Phrase-Based system using a system combination method based on confusion networks. We also present techniques used to preprocess the data before translation and to postprocess the translation output.

In this year's evaluation, we participated in the Arabic-to-English BTEC task, and the DIALOG task for translation between English and Chinese in both directions. Both the 1-best ASR hypotheses and the correct recognition results are translated for the DIALOG task.

This paper is organized as follows. Section 2 describes the systems we built for our participation. In Section 3 we describe different preprocessing and postprocessing techniques we used. Section 4 explains our experimental setup. We report the official results of our submitted systems in Section 5. In Section 6, we conclude and provide avenues for future work.

2. Translation Systems

In this section we describe translation systems we built in addition to the Phrase-Based and Hierarchical Phrase-Based baseline systems: a paraphrase Phrase-Based system, a Syntax Augmented Machine Translation (SAMT) system, a CCG augmented Hierarchical Phrase-Based system and system combination.

2.1. Hierarchical Phrase-Based System

SAMT was introduced in [3] as an extension to Hierarchical Phrase-Based Machine Translation. Hierarchical Phrase-Based MT systems are based on synchronous Context Free Grammar. Synchronous translation rules are extracted from the training corpus automatically according to the method described in [2]. The rules take the form:

$$X \rightarrow < \alpha, \beta, \sim >$$

where X is a nonterminal, α and β are both strings of terminals and nonterminals, and $\sim$ is a one-to-one correspondence between nonterminal occurrences in α and nonterminal occurrences in β. The following is an example of the synchronous CFG extracted from the Mandarin-English sentence pair (Aozhou shi yu Beihan you bangjiao de shaoshu guojia zhiyi, Australia is one of the few countries that have diplomatic relations with North Korea) [2]:

$$X \rightarrow <\ yu\ X_1\ you\ X_2\ ,\ have\ X_2\ with\ X_1\ >$$

$$X \rightarrow <\ X_1\ de\ X_2\ ,\ the\ X_2\ that\ X_1\ >$$

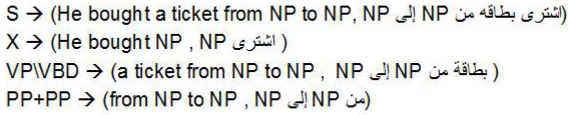
Nonterminals in hierarchical rules act as placeholders that can be replaced with other phrases. We can see from the previous examples that nonterminals in pure hierarchical rules do not have any syntactic annotation. In fact, hierarchical rules try to capture the hierarchical nature of the language but without any syntactic annotation. This means that no syntactic constraints are imposed on target phrase replacements during translation, which may lead to ungrammatical translations.

2.2. Syntax Augmented Machine Translation System

SAMT tries to enhance the quality of the pure Hierarchical Phrase-Based system by attaching syntactic labels to nonterminals in hierarchical rules. Those labels are extracted from the parse tree of the target-side sentence and act as syntactic constraint on phrases replacing nonterminals in hierarchical rules, which produces more grammatical translations. SAMT rules are extracted according to the following steps:

- Each sentence in the target-side is assigned a constituent grammar-based parse tree.
- Phrase pairs are extracted from the parallel corpus according to the method presented in [1].
- A syntactic label is assigned to each of the previously extracted phrase pairs. This syntactic label corresponds to the syntactic constituent in the parse tree that covers the target phrase. In case the target phrase is not fully covered by a constituent in the parse tree, the phrase is assigned an extended category of the form C1+C2, C1/C2, or C2\C1, indicating that the phrase pair's target-side spans two adjacent syntactic categories (e.g., she went: NP+V), a partial syntactic category C1 missing a C2 to the right (e.g., the great: NP/NN), or a partial C1 missing a C2 to the left (e.g. great wall: DT\NP), respectively.
- Hierarchical rules are extracted from syntactic-labeled phrases according to the method described in [2].

Figure 1 shows a sample of the hierarchical rules extracted from the English sentence (He bought a ticket from Ankara to Dublin) and its Arabic source sentence. Figure 2 illustrates the syntax tree associated with the English sentence (He bought a ticket from Ankara to Dublin) along with its aligned Arabic source sentence.



S → (He bought a ticket from NP to NP, NP اشترى بطاقه من NP إلى NP)
X → (He bought NP , NP اشترى)
VP\VBD → (a ticket from NP to NP , NP بطاقة من NP إلى NP)
PP+PP → (from NP to NP , NP من NP إلى NP)

Figure 1: Some hierarchical rules extracted from the English sentence (He bought a ticket from Ankara to Dublin) and its aligned Arabic source sentence according to SAMT method.

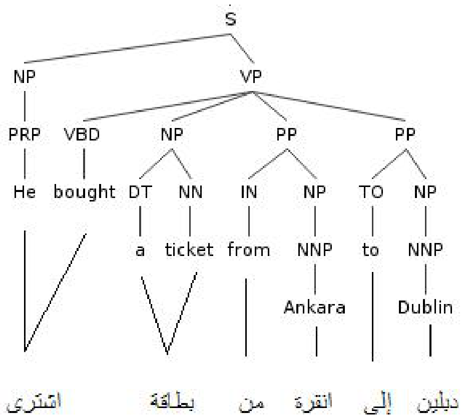


Figure 2: Parse tree of the English sentence (He bought a ticket from Ankara to Dublin) along with its aligned Arabic source sentence.

In our experiments, we use Moses SAMT4[1] to build our SAMT system. The Moses SAMT4 nonterminal labeling method is the same as explained above, but in addition to the SAMT basic operators (+,\, /), Moses SAMT4 uses three additional operators (++, //, \\):

- ++ to combine two adjacent constituents C1 and C2 which are not children of the same parent.
- // for example C1//C2 indicates partial syntactic category C1 missing a C2 to the right. C2 is not a direct child of C1.
- \\ for example C1\\C2 indicates partial syntactic category C1 missing a C2 to the left. C2 is not a direct child of C1.

2.3. CCG Augmented Hierarchical Phrase-Based System

2.3.1. Combinatory Categorial Grammar

CCG [4] is a grammar formalism which consists of a lexicon which pairs words with lexical categories (supertags) and a set of combinatory rules which specify how the categories are combined. A supertag is a rich syntactic description that specifies the local syntactic context of the word in the form of a set of arguments. Most of the CCG grammar is contained in the lexicon, so CCG has simple combinatory rules

[1] http://www.statmt.org/moses/?n=Moses.SyntaxTutorial

to combine CCG supertags. CCG categories are divided into atomic and complex categories. Examples of atomic categories are: S(sentence), N (Noun), NP (noun phrase). Complex categories such as S\NP and (S\NP)/NP are functions which specify the type and directionality of their arguments (primitive or complex categories) and the type of their result (primitive or complex category). Complex categories come in the following formats:

- X\Y is a functor X which takes as an argument the category Y to its left (which might be a primitive or complex category) and the result is the category X (which might also be a primitive or complex category).
- X/Y is a functor which takes as an argument the category Y to its right (which might be a primitive or complex category) and the result is the category X (which might also be a primitive or complex category).

For example the lexical category for the verb (eat) in the sentence (I eat) is S\NP which means that this category is expecting an NP (which plays the role of the subject in this case) to its left and the result of this category when an NP comes to its left is a sentence (S). By contrast, in the sentence (I eat an apple) the lexical category assigned to the verb eat in this case is (S\NP)/NP which means that it expects an NP to its left (which plays the role of the subject) and another NP to its right (which plays the role of the object), and the result of this category when all of its arguments are present is a whole sentence (S). Thus the complex lexical category (S\NP)/NP represents a transitive verb while the lexical category (S\NP) represents an intransitive verb.

CCG has a simple set of combinatory operators that are used to combine supertags. Those operators are divided into:

- Application operators: They are divided into forward and backward application operators. The forward operators combine a category X/Y with category Y to its right and the result is category X. Backward application: combines a category X\Y with category Y to its left and the result is category X.
- Composition operators: They are divided into forward and backward composition operators. The forward composition operators combine category X/Y with category Y/Z and the result is category X/Z. Backward composition operator combines category Y\Z with category X\Y category and the result is category X\Z.
- Type raising operators: Type raising operators turn arguments into functions over functions over such arguments. There are two types of type raising operators: forward and backward type rasing operators. Forward type raising operators transform category X into T(/X\T). Backward type rasing operators transform category X into T\(X/T), where T is a complex or primitive CCG category.

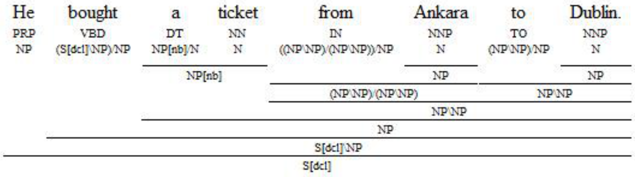
2.3.2. CCG-based Nonterminal Labeling

While SAMT augments the Hierarchical Phrase-Based system with constituent grammar-based syntax, we adopt the same approach followed by SAMT for labeling nonterminals in hierarchical rules, but using CCG instead of constituent grammar. CCG provides many advantages for use in HPB nonterminal labeling in comparison with constituent grammar. First, CCG has flexible structures that results from the ability to combine CCG supertags using simple combinatory operators. This flexibility enables CCG to assign a CCG supertag to a phrase that does not correspond to a syntactic constituent by simply combining the supertags of its words using CCG combinatory operators. This is very important for SMT systems as most of the phrases extracted in SAMT systems do not necessarily correspond to a syntactic constituent. Thus, while SAMT fails in labeling many phrases which do not correspond to a syntactic constituent because of rigid constituent grammar structures, CCG succeeds more on nonterminal labeling. Second, CCG supertags express rich information about the word or phrase dependents and its syntactic context. SAMT extracted labels express this information less accurately as they do not necessarily express the true dependents of the word or phrase. Therefore, CCG labeled nonterminals in hierarchical rules will act as syntax-rich placeholders which will be replaced by the phrases that best fit their syntactic context, and this can lead to more grammatical translations. Third, CCG parsing is more efficient in comparison with constituent grammar parsing. Because most of the CCG grammar is contained in the lexicon, assigning supertags to the words in the sentence is considered “almost parsing” [5]. After that, the CCG parser is only required to combine those supertags using CCG simple combinatory operators.

The extraction of CCG labeled hierarchical rules is done similarly to the SAMT approach:

- First, each target-side sentence from the parallel corpus is supertagged by assigning the best sequence of CCG supertags to its words.
- Next, phrase pairs are extracted from the parallel corpus according to the method presented in [1].
- Then, each extracted phrase pair is a assigned a CCG supertag that results from combining the supertags of the target phrase words. In case no CCG supertag can be assigned to the phrase, a general X label is assigned to it.
- Finally, hierarchical rules are extracted from sentence-pairs according to the method specified in [2]

Figure 3 shows the CCG parse tree of the English sentence (He bought a ticket from Ankara to Dublin) in addition to some of the hierarchical rules extracted from it and its aligned Arabic source sentence.


He bought a ticket from Ankara to Dublin.
PRP VBD DT NN IN NNP TO NNP
NP (S[dcl]\NP)/NP NP[nb]/N N ((NP\NP)/(NP\NP))/NP N (NP\NP)/NP N
NP[nb]
NP
NP
(NP\NP)/(NP\NP)
NP\NP
NP\NP
NP
S[dcl]\NP
S[dcl]

S → (He bought NP, اشترى NP)
S → (He bought NP from NP to NP, اشترى NP من NP إلى NP)
S → (He bought a ticket, اشترى بطاقة)
NP\NP → (from Ankara to Dublin , من أنقرة إلى دبلين)
NP\NP → (from NP to NP , من NP إلى NP)
S\NP → (He bought , اشترى)

Figure 3: CCG derivation tree along with a sample of hierarchical rules extracted from the English sentence (He bought a ticket from Ankara to Dublin) and its aligned Arabic source sentence according to our CCG-based approach.

2.4. System Combination

We combine the output of three hierarchical systems: a pure Hierarchical Phrase-Based system, a Syntax Augmented Machine Translation system (SAMT) and a CCG augmented Hierarchical Phrase-Based system. We use MANY [6] which is an open source tool for MT system combination based on the decoding of a lattice made of several confusion networks. System combination is done according to the following steps:

- 1-best hypotheses from all MT systems are aligned in order to build confusion networks.
- All confusion networks are connected into a single lattice.
- A language model is used to decode the resulting lattice and the best hypothesis is generated.

MANY has the following parameters:

- TERp costs: the costs of insertion, deletion, substitution, shift, synonym and stem
- System priors
- Fudge factor
- Null-arc penalty
- Length penalty

We use Condor [7] to tune those parameters on a development data. Condor is an optimizer which tries to minimize an objective function. The objective function in our case is the negative of the BLEU score of the whole system combination output.

2.5. Paraphrase MT System for English-Chinese

To overcome the limited amount of training resources in the IWSLT evaluation campaign, we utilized source-language paraphrases to build a contrastive MT system [8] for the DIALOG English-Chinese task. The source language paraphrases are generated from the parallel corpora based on the algorithm in [9] to enhance the baseline Phrase-Based system for the DIALOG English-Chinese task. For the the development and test sets, paraphrase options are presented as source-side lattices, and the probabilities on edges in the lattices are formed to penalize on the paths going through paraphrase options as in [8]. MERT [10] is utilized to tune the weights of the paraphrase probabilities from the development set, and decoding on the test set is carried out on lattice inputs. The purpose of this contrastive deployment is to validate the effectiveness of the paraphrase systems in the spoken language translation task using limited training resources, both for the correct recognition results (CRR) and ASR 1-best case. The comparison is reported in Section 5.

3. Data Preprocessing and Postprocessing

In this section we present the techniques we used for preprocessing data before translation and postprocessing it after translation, namely: Arabic preprocessing by morphological segmentation, Chinese segmentation and numbers handling, punctuation restoration for Chinese and English DIALOG task test data before translation and Case restoration for English data after translation.

3.1. Arabic Segmentation

Arabic is a morphologically rich language. Information such as gender, number, tense and aspect is expressed as clitics attached to the Arabic word. In addition, some words such as prepositions, conjunctions and possessive pronouns attach to other words. This hugely increases the number of different forms of Arabic words and as a result poses a word sparsity problem for SMT systems.

Several methods have been devised to solve the Arabic word sparsity problem in SMT systems. The method we adopt in our experiment reduces Arabic word sparsity by morphologically segmenting Arabic data as a preprocessing step before translation [11]. There are different levels for Arabic morphological segmentation. Each level segments different types of clitics. The deeper the segmentation, the more complex the morphological analysis and the less sparse the data become. The effect of each level of segmentation on translation performance depends of the size of the training data [12]. Small amounts of training data require deeper segmentation while large training sizes require only simple segmentation. In order to choose the best segmentation level for the IWSLT experiments, we examined the effect of different segmentation levels on the IWSLT 2008 test set. We used MADA (Morphological Analysis and Disambiguation for Arabic) [13], which can be adjusted to segment the words

Scheme	BLEU
Simple tok	45.44
D2	50.62
D3	**53.20**
TB	52.75

Table 1: BLEU scores of Arabic-English Hierarchical Phrase-Based system using different Arabic segmentation schemes on IWSLT 2008 test data.

according to different segmentation levels called Schemes. We examined three different segmentation schemes:

- Simple tokenization: separation of punctuation marks and numbers from words.
- D2 scheme: separation of conjunctions clitics (w+ and f+) and class of particles (l+, b+, k+, and s+).
- D3 scheme: separation of the same clitics as in D2, in addition to the definite article (AL+) and pronominal enclitics.
- TB scheme: separation of the same clitics as in D3, except for the definite article (AL+) and the future particle (s+).

Table 1 shows BLEU [14] scores for our Arabic-English Hierarchical Phrase-Based system on IWSLT 2008 test data using different segmentation levels. Clearly, deeper segmentation levels achieve better performance. The D3 scheme achieved the best score among other schemes. That is why we chose this scheme to preprocess Arabic data in our Arabic-to-English experiments.

3.2. Chinese preprocessing and postprocessing

Chinese training, development and test sets are preprocessed to better suit our MT systems. The following procedures are carried out in this section:

- Punctuation restoration for ASR 1-best inputs.
- Word re-segmentation using the ICTCLAS tool[2] for better segmentation.
- Using heuristic rules to adjust the segmentation results for Chinese numbers.
- Conversion of punctuation marks, numbers and Latin letters from Chinese form into Latin form.

After MT, punctuation marks, numbers and Latin letters are converted back from the Latin form into the Chinese form as a postprocessing step.

3.3. Case and Punctuation Restoration

IWSLT evaluation is done on punctuated and true-case data. To reduce word sparsity, we train our models on lower case data and then restore the case information after translation as a postprocessing step. As the test data of the DIALOG task does not contain punctuation, we have two choices: either to train our models on non-punctuated data and then restore punctuation marks after translation, or to train our models on punctuated data and restore punctuation before translation as a preprocessing step. In order to obtain better alignments, we choose the second method.

For case restoration, we treat it as a translation task; we train a phrase-based translation model on training data that consists of lower case English data on the source side and its true-case original data on the target-side. Therefore, restoring case is done by translating the lower case data using this model to true-case data.

As for punctuation restoration, we used the hidden-ngram tool in the SRILM toolkit [15] to insert punctuation marks in the text before translation. We train the language model used by this hidden-ngram tool on the training data.

4. Experimental Setup

4.1. Data Sets

In our experiments, we used the data provided by the IWSLT evaluation campaign. For each task, we use one of the provided sets for MERT tunning, another set for system combination tuning (DIALOG Chinese-English task, BTEC Arabic-English Task) and a third set for our internal evaluations. Then all of the other sets are merged with the provided training data.

All the English data used in our experiments is lower cased and tokenized. For the SAMT system, we parse the English side of the training corpus using the Berkeley Parser.[3] We use CCG parser from C&C tools[4] to parse the training data for our CCG augmented hierarchical system experiments. Sentence pairs whose English side fails to parse are removed from the training data. That is why the size of the training data of our SAMT and CCG augmented HPB systems is less than the size of the training data of the HPB system.

Table 2 shows that size of the training data used to build each of our systems. Table 3 shows the data sets used for our internal testing in addition to MERT and system combination tuning for each task.

4.2. Machine Translation Systems

For experiments that use the Phrase-Based model, we use the Moses Phrase-Based Decoder [16] with maximum phrase length=12. Hierarchical Phrase-Based systems are built us-

[2] http://www.ictclas.org/index.html

[3] http://code.google.com/p/berkeleyparser/

[4] http://svn.ask.it.usyd.edu.au/trac/candc/

System	MERT devset	Syscomb devset	Testset
BTEC Arabic-English	IWSLT07 testset	IWSLT07 testset	IWSLT08 testset
DIALOG Chinese-English	IWSLT08 BTEC testset	IWSLT08 DIALOG testset	IWSLT07 BTEC testset
DIALOG English-Chinese	IWSLT08 DIALOG testset	-	IWSLT09 devset

Table 3: Data sets used for testing and MERT tuning in addition to system combination tuning in each task.

System	Data size
AE HPB	21484
AE SAMT	21423
AE CCG	20376
CE HPB	63234
CE SAMT	63084
CE CCG	60513
EC PB	71725
EC Paraphrase	71725

Table 2: Training data size used to build each system in each task. AE indicates Arabic to English, CE indicates Chinese to English, EC indicates English to Chinese.

ing Moses Chart-Decoder.[5] The SAMT4 scheme in the Moses Chart-Decoder is used to build our SAMT system. For all our hierarchical systems, maximum phrase length is set to 12 and maximum rule span is set to 15. Rules extracted contain up to 2 nonterminals. The GIZA++ toolkit[6] is used to perform word alignment and "grow-diag-final" refinement method is adopted [1]. Minimum error rate training [10] is performed to tune all our SMT systems. The language model in all experiments is 5-gram trained on the target side from the parallel corpus using the SRILM toolkit [15] with modified Kneser-Ney smoothing [17].

5. Experiments Results

In the following subsections, we report the results of our experiments on the IWSLT 2010 test set for BTEC Arabic-English, DIALOG Chinese-English, and DIALOG English-Chinese Tasks.

5.1. BTEC Task Arabic-English

Table 4 shows the official evaluation results of the Hierarchical Phrase-Based systems in addition to their combination on true-cased punctuated translation of the test data. We can see that the Hierarchical Phrase-Based system achieved the highest BLEU score. System combination in this case did not result in any improvement. This might be due to inconsistencies between the data set used for system combination tuning and the evaluation test set. Syntax augmentation to the Hierarchical Phrase-Based system (SAMT4, CCG augmented HPB) did not result in an improvement either. BLEU, METEOR [7] and TER [8] scores show that all systems seem to have similar performance.

System	BLEU	METEOR	TER
HPB	**46.15**	73.82	32.47
CCG	45.30	**73.98**	33.88
SAMT4	46.06	73.74	**32.37**
Syscomb	46.11	72.75	32.62

Table 4: Official results of single systems and multiple system combination for BTEC Arabic-English task

5.2. DIALOG Task Chinese-English

Table 5 shows the official evaluation results of our Chinese-English hierarchical systems and their system combination for correct recognition results (CRR). The CCG augmented Hierarchical Phrase-Based system achieved the best BLEU score beating the pure Hierarchical Phrase-Based system by 0.63 absolute BLEU points which accounts for 4.63% relative improvement. SAMT4 system comes in second place behind the CCG-based HPB system by 0.46 absolute BLEU points which accounts for 3.34% relative improvement. This improvement comes despite the fact that the CCG-based HPB system training data has 2721 fewer sentence pairs in comparison with the Hierarchial Phrase-Based system, which corresponds to 4.3% of the total number of sentence pairs in the training data. Those sentences are the sentences whose English part failed to parse with the CCG parser. SAMT4 in turn achieved a better score than the pure Hierarchical Phrase-Based system by 0.17 absolute BLEU points which corresponds to 1.25% relative improvement. System combination in this case did not improve the performance of the combined systems.

Table 6 shows the official results of evaluating the 1-best ASR output of the Chinese-English hierarchical systems and their combination. Similar to the correct recognition results, CCG-based system achieved the best BLEU score. The SAMT4 system came in second place followed by the Hierarchical Phrase-Based system. However, the differences in performance between those systems are smaller in the case of ASR output in comparison with the CRR results. The CCG augmented HPB system outperformed the Hierarchical

[5] http://www.statmt.org/moses/?n=Moses.SyntaxTutorial

[6] http://fjoch.com/GIZA++.html

[7] http://www.cs.cmu.edu/ãlavie/METEOR/

[8] http://www.cs.umd.edu/s̃nover/tercom/

System	BLEU	METEOR	TER
HPB	13.58	40.90	**64.67**
CCG	**14.21**	40.70	65.86
SAMT4	13.75	40.87	65.20
Syscomb	13.96	**41.39**	65.48

Table 5: Official results of evaluating correct recognition results (CRR) of single systems and multiple system combination for DIALOG Chinese-English task

System	BLEU	METEOR	TER
HPB	12.79	38.92	**65.65**
CCG	**12.96**	39.09	67.23
SAMT4	12.86	39.16	66.40
Syscomb	12.69	**39.67**	66.65

Table 6: Official results of 1-best ASR output of single systems and multiple system combination for DIALOG Chinese-English task

Phrase-Based system by 0.17 absolute BLEU points which corresponds to 1.32% relative improvement. SAMT4 performed so close to the CCG augmented HPB with only 0.10 absolute BLEU points, which accounts for 0.77% relative difference between the two systems. In general, we observe that all systems perform better under the correct recognition condition than under the ASR condition. This is because we trained our models on training data that consists merely of correct recognition data.

5.3. DIALOG Task English-Chinese

Table 7 shows the official evaluation of paraphrase system and the Phrase-Based system correct recognition results (CRR) output for English-Chinese DIALOG task. The results show that the paraphrase system outperformed the Phrase-Based system on BLEU, METEOR and TER metrics. The paraphrase system achieved 0.89 absolute BLEU points which corresponds to a 4.42% relative improvement over the Phrase-Based system baseline.

Table 8 shows the official evaluation of our paraphrase system and the Phrase-Based system 1-best ASR output for the English-Chinese DIALOG task. Similar to the correct recognition results, the paraphrase system outperformed the Phrase-Based baseline system on BLEU, METEOR and TER metrics. The Paraphrase system outperformed the Phrase-Based baseline system by 1.10 absolute BLEU points which corresponds to a 6.35% relative improvement. This demonstrates the ability of paraphrase systems to improve the performance of the Phrase-Based baseline system under both correct recognition and ASR conditions of the spoken language domain and using limited training resources. Analogous to the Chinese-English task, we remark that systems under the ASR condition show poorer performance for the same reason.

System	BLEU	METEOR	TER
PB	20.13	45.89	69.85
Paraphrase	**21.02**	**46.21**	**66.41**

Table 7: Official results of evaluating correct recognition results (CRR) output of Phrase-Based system and Paraphrase system for DIALOG English-Chinese task

System	BLEU	METEOR	TER
PB	17.32	41.41	78.05
Paraphrase	**18.42**	**42.34**	**74.07**

Table 8: Official results of evaluating ASR-1 output of Phrase-Based system and paraphrase system for DIALOG English-Chinese task

6. Conclusion

In this paper we described the MT systems we built for our participation at the IWSLT 2010 evaluation campaign in the BTEC Arabic-English and DIALOG Chinese-English and English-Chinese tasks. We tried two syntax augmentation methods to the Hierarchical Phrase-Based system using two different types of grammar formalisms: SAMT based on constituent grammar, and a CCG augmented Hierarchical Phrase-Based system based on CCG. Experiments showed that syntax improved the performance of the Chinese-to-English Hierarchical Phrase-Based system under both correct recognition and ASR conditions. For Arabic-to-English translation, the syntax-augmented hierarchical systems showed no improvement. This might be due to the difference in size between the Arabic-English training data and the Chinese-English training data. In fact, the Chinese-English training data is about 3 times larger than the Arabic-English training data. This might affect the performance of syntax augmented systems as they are sensitive to data sparsity which increases in effect as the size of the training data decreases. In addition, our experiments showed that the paraphrase system outperformed the Phrase-Based baseline system for English-to-Chinese translation under both ASR and correct recognition conditions. This demonstrates the ability of this method to improve performance using limited training resources and in the domain of spoken language not only under the correct recognition condition but also under the automatic speech recognition condition.

We also tried to improve the performance of our Hierarchical systems by using system combination based on confusion networks. However, our experiments show that no performance gain was achieved for system combination over the combined systems. This might be due to inconsistencies between the development set used for tuning system combination parameters and the evaluation set.

7. Acknowledgements

This research is supported by Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (http://www.cngl.ie) at Dublin City University.

8. References

[1] P. Koehn, F. Och, and D. Marcu. 2003. *Statistical phrase-based translation*. In Proceedings of the 2003 Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics, Edmonton, AB, Canada, 2003, pp. 48‘54.

[2] D. Chiang. 2005. *A hierarchical phrase-based model for statistical machine translation*. In ACL-2005: 43rd Annual meeting of the Association for Computational Linguistics, Ann Arbor, MI, pp. 263 - 270.

[3] A. Zollmann, A. Venugopal. 2006. *Syntax augmented machine translation via chart parsing*. HLT-NAACL 2006: Proceedings of the Workshop on Statistical Machine Translation, New York, NY, USA, pp. 138-141.

[4] M. Steedman. 2000 *The Syntactic Process*. MIT Press, Cambridge, MA.

[5] S. Bangalore and A. Joshi. 1999. *Supertagging: An Approach to Almost Parsing.* Computational Linguistics 25(2):237-265, 1999.

[6] L. Barrault. 2010. *MANY : Open Source Machine Translation System Combination.* Prague Bulletin of Mathematical Linguistics, Special Issue on Open Source Tools for Machine Translation, 93:147-155.

[7] F. V. Berghen, and H. Bersini. 2005. *CONDOR, a new parallel, constrained extension of Powell's UOBYQA algorithm: Experimental results and comparison with the DFO algorithm.* Journal of Computational and Applied Mathematics,2005, 181:157-175.

[8] J. Du, J. Jiang and A. Way. 2010. *Facilitating Translation Using Source Language Paraphrase Lattices.* In Proceedings of the Conference on Empirical Methods in Natural Language Processing. (EMNLP 2010). Cambridge, MA, pp. 420-429.

[9] C. Bannard and C. Callison-Burch. 2005. *Paraphrasing with bilingual parallel corpora.* In 43rd Annual Meeting of the Association for Computational Linguistics, Ann Arbor, MI, pp. 597-604.

[10] F. Och. 2003. *Minimum Error Rate Training in Statistical Machine Translation*. In ACL-2003: 41st Annual Meeting of the Association for Computational Linguistics, pp. 160-167, Sapporo, Japan.

[11] N. Habash and F. Sadat. 2006. *Arabic preprocessing schemes for statistical machine translation.* Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL, New York, NY, USA, pp. 49-52.

[12] F. Sadat and N. Habash. 2006. *Combination of Arabic preprocessing schemes for statistical machine translation.* Coling-ACL 2006: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics,Sydney, pp.1-8.

[13] N. Habash, O. Rambow and R. Roth. 2009. *MADA+TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging, Stemming and Lemmatization.* In Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Cairo, Egypt, pp.242-245.

[14] K. Papineni, S. Roukos, T. Ward and W-J. Zhu. 2002. *BLEU: A Method for Automatic Evaluation of Machine Translation*. In Proceedings of 40th Annual meeting of the Association for Computational Linguistics, Philadelphia, PA., pp.311-318.

[15] A. Stolcke. 2002. *SRILM – an Extensible Language Modeling Toolkit.* In Proceedings of the International Conference on Spoken Language Processing, Denver, Colorado, USA, volume 2, pp. 901-904.

[16] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, E. Herbst. 2007. *Moses: open source toolkit for statistical machine translation*. ACL 2007: proceedings of demo and poster sessions, Prague, Czech Republic, pp. 177-180.

[17] R. Kneser and H. Ney. 1995. *Improved Backing-Off for M-Gram Language Modeling.* In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP),Detroit, Michigan, USA, Vol. 1, pp. 181-184.