The DCU Machine Translation Systems for IWSLT 2011

Pratyush Banerjee, Hala Almaghout, Sudip Naskar, Johann Roturier,[1]
Jie Jiang,[2] *Andy Way,*[2] *Josef van Genabith*

CNGL, School of Computing, Dublin City University, Dublin, Ireland
{pbanerjee, halmaghout, snaskar, josef}@computing.dcu.ie
[1] Symantec Limited, Dublin, Ireland
johann_roturier@symantec.com
[2]Applied Language Solutions, Delph, UK
{andy.way, jie.jiang}@appliedlanguage.com

Abstract

In this paper, we provide a description of the Dublin City University's (DCU) submissions in the IWSLT 2011 evaluation campaign.[1] We participated in the Arabic-English and Chinese-English Machine Translation(MT) track translation tasks. We use phrase-based statistical machine translation (PBSMT) models to create the baseline system. Due to the open-domain nature of the data to be translated, we use domain adaptation techniques to improve the quality of translation. Furthermore, we explore target-side syntactic augmentation for an Hierarchical Phrase-Based (HPB) SMT model. Combinatory Categorial Grammar (CCG) is used to extract labels for target-side phrases and non-terminals in the HPB system. Combining the domain adapted language models with the CCG-augmented HPB system gave us the best translations for both language pairs providing statistically significant improvements of 6.09 absolute BLEU points (25.94% relative) and 1.69 absolute BLEU points (15.89% relative) over the unadapted PBSMT baselines for the Arabic-English and Chinese-English language pairs, respectively.

1. Introduction

In this paper we describe the machine translation systems built for our participation in IWSLT 2011 evaluation campaign [1] for the Arabic-English (Ar–En) and Chinese-English (Zh–En) MT track translation tasks. We use different SMT models, ranging from standard phrase-based SMT models [2] to CCG-augmented hierarchical phrase-based models [3] to translate the test data provided. The open-domain nature of the data and the restricted size of the in-domain training corpora necessitated the use of domain adaptation techniques to improve translation quality.

The baseline system built for the task is a simple PBSMT system trained only on the 'in-domain' training data released as a part of the evaluation campaign. This training data comprised of both parallel and monolingual data from the TED Talks:[2] a collection of public speeches on a variety of topics. Out-of-domain data in the form of a parallel Multi-UN corpus[3] was also available to enrich the models trained on in-domain data. For domain-adaptation we enhanced the language models built on the TED corpus data with selected data from the UN corpus. Mixture adaptation [4] techniques were used to combine models from multiple sources weighted according to their fit with respect to the development set. The adapted language models provided an improvement of about 5.16 absolute (21.99% relative) BLEU points for Ar–En and 1.25 absolute (11.76% relative) BLEU points for Zh–En language pairs over the unadapted baseline.

Once the best performing adapted language models were identified, we tried to further boost the performance by providing the HPB SMT system with target-side syntactic information extracted using CCG resources [5]. We used CCG categories to label non-terminals in hierarchical rules. Different CCG-based labeling approaches were explored, each focussing on a different aspect of information reflected in CCG categories. The best performing system was a CCG-augmented HPB system for both language pairs providing a statistically significant improvement of 0.93 absolute BLEU points (3.25% relative) and 0.44 absolute BLEU points (3.7% relative) over the Ar–En and Zh–En mixture-adapted PBSMT baselines, respectively.

The paper is organized as follows: Section 2 provides a brief description of the different SMT models and adaptation techniques used in our experiments. Section 3 details our experimental setup with descriptions on the specific toolsets and data used. Section 4 provides the results of each set of experiments as well as analyses, followed by conclusion and future work in Section 5.

2. Translation Systems

This section focuses on the different translation techniques used in the experiments.

[1]http://iwslt2011.org

[2]http://www.ted.com/talks

[3]http://www.euromatrixplus.eu/downloads/35

2.1. Phrase-based SMT Systems

Phrase-based SMT systems [2] are the most commonly used technique in statistical machine translation nowadays. In this approach, source and target phrase pairs consistent with the word alignment are extracted from the parallel training data. Phrases in PBSMT are just contiguous chunks of text, and are not linguistically motivated. The extracted source-target phrase pairs along with their translation probabilities (computed from the same training data) are stored in a structure known as the 'phrase table'. During translation, an input sentence is split up into phrases and their corresponding translations are looked up from the phrase table to create a set of translated sentences in the target language. The target phrases in each such translation are subsequently reordered using a statistical re-ordering model that assigns a probability based on the orientation between a phrase and the previously translated phrase. A language model is further used for better fluency and grammaticality of the translation. The phrase translation probabilities along with reordering and language model probabilities are combined in a log-linear fashion to assign a score to each possible translation of an input sentence. Finally the best scoring translation is searched for by the decoding algorithm and is presented as the best translation for the corresponding input sentence. Formally this task can be expressed as in (1):

$$\hat{e} = \arg\max_{e} \sum_{i=1}^{K} \lambda_i h_i(f, e) \tag{1}$$

where, $h_i(f, e)$ denotes the different components for translating the source sentence f into the target sentence e. K is the number of components (or features) used and λ_i are the corresponding weights of the components. The Moses SMT system [6], which implements this particular model, was used for all our PBSMT translation experiments. Different component weights (λ_i) were estimated using a discriminative training method known as Minimum Error Rate Training (MERT) [7], on a held out development set (devset).

2.2. Mixture Adaptation of Language Models

Mixture Modelling [8], a well-established technique for combining multiple models, has been extensively used for language model adaptation in SMT [4]. This technique has also been used for adapting the translation model in SMT with limited success [9]. For the given task, since the size of the 'in-domain' data was not significantly large, we used 'suitable' subsets of data from the other available 'out-of-domain' corpora to enrich the models.

For a mixture adapted language model, the probability of an n-gram hw is given as in (2):

$$Pr_{mix}(w|h) = f^*_{mix}(w|h) + \lambda_{mix}(h) Pr_{mix}(w|\bar{h}) \tag{2}$$

where w is the current word, h is the corresponding history, f^*_{mix} is the mixture model discounted relative frequency, λ_{mix} indicates the mixture model zero-frequency estimate and $\bar{h}w$ is the lower order $n-1$ gram. The discounted frequency and zero-frequency estimates are defined as follows:

$$f^*_{mix}(w|h) = \sum_{i=1}^{k} \mu_i f^*_i(w|h) \tag{3}$$

$$\lambda_{mix}(h) = \sum_{i=1}^{k} \mu_i \lambda_i(h) \tag{4}$$

$$\lambda_i(h) = 1.0 - \sum_{w \in V} f^*_i(w|h) \tag{5}$$

where k is the number of language models which are being interpolated, μ_i the interpolation weights and V is the vocabulary of the specific language model. The interpolation weights are estimated using Expectation Maximization (EM) [10] over the log-likelihood in (6):

$$\sum_{t=1}^{N} log \sum_{i=1}^{k} \mu_i (f^*_i(w_t|h_t) + \lambda_i(h_t) Pr_{mix}(w_t|\bar{h_t})) \tag{6}$$

where the index t scans over all the n-grams in the training corpora. This mixture model was used to combine the 'in-domain' language model with an 'out-of-domain' one, with the mixture weights being estimated on the 'in-domain' training data by applying a cross-validation scheme. Further improvements on this mixture models were achieved using parameter tying to the most-recent context words [4].

2.3. Hierarchical Phrase-Based System

Hierarchical Phrase-Based (HPB) SMT [3] is a tree-based model which extracts a synchronous Context-Free Grammar (CFG) automatically from the training corpus. HPB SMT is based on phrases extracted according to the PB model [2]. Thus, HPB SMT tries to build upon the strengths of PB SMT and adds to it the ability to translate discontinuous phrases and learn phrase-reordering in hierarchical rules without a separate reordering model. HPB SMT uses hierarchical rules as a translation unit. These rules are rewrite rules with aligned pairs of right-hand sides, taking the following form:

$$X \rightarrow < \alpha, \beta, \sim > \tag{7}$$

where X is a non-terminal, α and β are both strings of terminals and non-terminals, and $\sim$ is a one-to-one correspondence between non-terminal occurrences in α and β. The following are examples of the hierarchical CFG rules extracted from the Chinese–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`) [3]:

$$X \rightarrow <\ yu\ X_1\ you\ X_2\ ,\ have\ X_2\ with\ X_1\ > \tag{8}$$

$$X \rightarrow <\ X_1\ de\ X_2\ ,\ the\ X_2\ that\ X_1\ > \tag{9}$$

Hierarchical rules are extracted from the training corpus by subtracting continuous phrase-pairs attested in the translation table recursively from longer phrases and replacing them with the non-terminal symbol X. Non-terminals in hierarchical rules act as placeholders that are replaced with other phrases during translation in a bottom-up fashion. Hierarchical rules are extracted from the training corpus without using any syntactic information. As the resulting system is syntactically unaware, the HPB SMT system can produce ungrammatical translations. Therefore, several approaches have tried to provide the HPB SMT system with syntactic information. Syntax augmented Machine Translation (SAMT) [11] uses target-side phrase-structure grammar syntactic trees to label non-terminals in hierarchical rules. These non-terminal labels represent syntactic constraints imposed on target phrase replacements during translation aiming to produce more grammatical translations.

2.4. CCG-augmented HPB System

Following the SAMT approach, CCG-augmented HPB SMT [12] uses CCG [5] to label non-terminals. CCG has distinct advantages over phrase-structure grammar in the general SMT context, particularly in extracting non-terminal labels in HPB SMT. This section gives a brief introduction to CCG followed by a description of the approach of extracting non-terminal labels using the same.

2.4.1. Combinatory Categorial Grammar

CCG [5] is a grammar formalism which consists of a lexicon that 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, that is why CCG has simpler combinatory rules in comparison to CFG production rules.

CCG categories are divided into atomic and complex categories. Examples of atomic categories are: S (sentence), N (noun), NP (noun phrase), etc. Complex categories such as S\NP and (S\NP)/NP are functions which specify the type and directionality of their arguments and results. Complex categories have the following formats:

- X\Y is a functor which takes as an argument the category Y to its left and the result is the category X. Both X and Y could be primitives or complex categories.
- X/Y is a functor which takes as an argument the category Y to its right resulting in the category X.

Representing CCG categories as functors and arguments reflects explicitly the dependents and local context of the word/phrase. For example, the lexical category of the verb *read* in the sentence *I read* is S\NP, which means that this category needs an NP (subject) as the left argument and the result of this category with an NP to its left is a sentence S. By contrast, in the sentence *I read a book*, the lexical category assigned to the verb *read* is (S\NP)/NP, meaning that it needs an NP as a left argument (subject) and another NP as a right argument (object), and the result is a whole sentence S.

2.4.2. CCG-based non-terminal Labelling

CCG provides many advantages when using it in SMT in comparison with phrase-structure grammar. Firstly, CCG has more flexible structures in comparison with phrase-structure grammar. This flexibility results from the ability to combine CCG supertags using simple combinatory operators, which makes it possible to assign a CCG category to a phrase that does not represent a traditional constituent in phrase-structure grammar. This is very important for SMT systems as the power of SMT lies in using statistically extracted phrases which do not necessarily correspond to syntactic constituents. Secondly, CCG categories reflect rich information about the syntactic structure to which the word/phrase belongs at the lexical level without the need to build a full parse tree for the sentence. Thirdly, CCG parsing is more efficient in comparison to phrase-structure grammar parsing. Because most of the CCG grammar is contained in the lexicon, the process of supertagging, which is to assign supertags (i.e. complex CCG categories) to the words in a sentence, is considered "almost parsing" [13]. After supertagging, the CCG parser is only required to combine the supertags using CCG simple combinatory operators. For the aforementioned reasons, CCG is considered more suitable to be used in SMT than phrase-structure grammar.

Attaching CCG categories to non-terminals in hierarchical rules is done in a way similar to that of 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 PBSMT phrase extraction method [2].
- Then, each phrase pair is assigned a CCG category that results from combining the supertags of the words of the target-side phrase using CCG combinatory operators. In case phrase parsing fails to find a single CCG category for the phrase, a general X label is assigned to the phrase.
- Finally, hierarchical rules are extracted from sentence-pairs according to the same basic HPB SMT rule extraction method [3].

During translation in the CCG-augmented HPB system, only phrases which have a label matching the label of a non-terminal are allowed to replace the same. This way non-terminal labels act as syntactic constraints on phrases replacing non-terminals during translation, driving the replacement process towards producing more grammatical translations.

Using CCG categories to label non-terminals in HPB rules can produce better translation quality and smaller trans-

lation models in comparison with SAMT [12]. CCG non-terminal labels are less sparse and represent richer and more accurate syntactic constraints compared to SAMT non-terminal labels [12].

2.4.3. Simplifying CCG non-terminal Labels

Despite of the advantages of using CCG categories to label non-terminals in the HPB system compared with SAMT labels, richness of CCG categories still leads to a large number of different non-terminal labels. This causes fragmentation of rule probabilities and consequently affects translation quality negatively. A CCG category C takes the form of C=(T\L)/R where L represents the left argument category, R the right argument category, and T the resulting category. Each of these constituent categories might be atomic or complex. Furthermore, some atomic CCG categories have features expressed between brackets which describe certain syntactic information. For example, the atomic category S might have a feature attached to it which distinguishes types of sentences such as declarative S[dcl] or wh-question S[wq]. All the additional information represented in a single CCG category increases the number of different CCG categories and leads to label sparsity problem. In order to address this problem, we simplify CCG non-terminal labels by reducing the amount of the information represented in them using the following approaches [14]:

- **Feature-dropped CCG labels**: these labels are extracted from CCG categories by dropping the syntactic features attached to atomic categories from the label representation. For example, if a phrase has a CCG category S[dcl]/NP, then its feature-dropped CCG label is S/NP.
- **CCG Contextual Labels**: in a CCG contextual label, only left and right argument categories are used in the label representation whereas the resulting category (i.e. the functor) is dropped from the label representation. The resulting CCG contextual label takes the form L_R. If any of the argument categories is missing, an X symbol is used in its place. For example, if a phrase has a CCG category (S\NP)/(S\NP), this means that it has NP as a left argument category while it has S\NP as a right argument category. Therefore, its CCG contextual label is NP_S\NP, which combines the left and right arguments in one label. In another example, if a phrase has a category NP\NP, then its CCG contextual label is NP_X.
- **Feature-dropped CCG Contextual Labels**: these labels are extracted from CCG contextual labels explained above by dropping syntactic features from the label representation. For example, if a phrase has a CCG category (S\NP[nb])/NP, then its feature-dropped CCG contextual label is NP_NP.

The above simplification methods reduce the total number of different CCG-based non-terminal labels which reduces label sparsity and lessens rule probability fragmentation. This comes of course at the expense of the accuracy of the syntactic constraints imposed on phrases during translation, which affects the grammaticality of the output. Our experiments will show the effects of this trade-off between label accuracy and sparsity.

3. Experimental Setups

This section details the setup for the different experiments. We also provide a brief account of the different tools and datasets used along with the preprocessing and postprocessing procedures employed.

3.1. Tools and Datasets

For our PBSMT-based translation experiments we used OpenMaTrEx [15], an open source SMT system which provides a wrapper around the standard log-linear phrase-based SMT system Moses [6]. Word alignment was performed using Giza++ [16]. The phrase and the reordering tables were built on the word alignments using the Moses training script. The feature weights for the log-linear combination of the feature functions were tuned using Minimum Error Rate Training (MERT) [7] on the devset with respect to BLEU [17]. We used 5-gram language models in all our experiments created using the IRSTLM language modelling toolkit [18] using Modified Kneser-Ney smoothing [19]. Mixture adaptation of language models mentioned in Section 2.2 was also performed using the features of the IRSTLM toolkit. Results of translations in every phase of our experiments were evaluated using BLEU, METEOR [20] and TER [21] metrics.

Table 1: *Number of Sentences for bilingual and monolingual data sets*

Data Set	**Ar–En**	**Zh–En**
TED parallel	90,379	106,776
Multi-UN	5,231,931	5,624,637
Development Set	934	934
Test Set	1,664	1,664
TED Monolingual	125,948	
Multi-UN Monolingual	5,796,505	

The datasets used for the experiments included the specific datasets released by the IWSLT 2011 evaluation campaign. The primary bi-lingual training data comprised of a collection of public speech transcriptions on a variety of topics from TED Talks. The development data released for the task, comprised of both the IWSLT-2010[4] development and test sets. However, for experiments reported in this paper, the IWSLT-2010 development set and test sets were used for tuning and testing respectively. As an auxiliary out-of-domain source of bi-lingual training data, the Multi-UN corpus was also released. The monolingual data required to train lan-

[4] http://iwslt2010.fbk.eu/node/15

guage models also comprised of data from both Multi-UN and TED Talks. Table 1 shows the exact sentence counts of the different datasets used in the experiments.

3.2. Data Preprocessing and Postprocessing

Arabic being a morphologically rich language, has many different surface forms of words with same root. This phenomenon poses a data sparsity problem for SMT systems. In order to reduce data sparsity, we segment the Arabic data morphologically before training. The Arabic data is segmented according to the D3 segmentation scheme using MADA (Morphological Analysis and Disambiguation for Arabic).[5] For all the available Chinese data, we segment the sentences to words using the Stanford Chinese Word Segmenter [22]. English data is lower-cased and tokenized in the preprocessing step.

After translation, we perform case restoration and detokenization for the English data. Case restoration, or truecasing is treated as a translation task. A simple phrase-based translation model is trained on aligned lower-case and truecase data to successfully achieve the task of true-casing.

3.3. PBSMT based Language Model Adaptation Experiments

As shown in Table 1, the size of the 'in-domain' TED training data is much smaller than the 'out-of-domain' Multi-UN training data. Since adding a significant amount of out-of-domain data to an in-domain corpus reduces the quality of translation for in-domain sentences [23], we decided to use only a part of the out-of-domain data to enhance the translation quality. In order to achieve this, we constructed a language model on the TED monolingual data and computed sentence-level perplexity score for all the sentences in Multi-UN, with respect to the TED language model. After sorting the sentences in the ascending order of the perplexity values, only sentences below a specific threshold were selected. This method provided us with the most 'TED-like' sentences from the Multi-UN corpora.

In order to decide which specific threshold gives us the best possible translation score, we experimented with multiple sets of 'selected' Multi-UN data corresponding to different thresholds. Finally we selected the particular threshold which gave us the best improvement over the standard PBSMT baseline. Since the range of the perplexity values for the Multi-UN corpus was huge, we used a simple heuristic of keeping the number of selected sentences from Multi-UN corpora less than the number of available training sentences in the TED corpus. This heuristic enabled us to keep the number of such experiments manageable by providing an upper bound on the perplexity value. The lower bound was manually decided on the basis of the number of sentences in the selection.

Once the range was decided, for each perplexity value in the range, we created a set of selected sentences from the Multi-UN corpora. Each such set was then combined with the TED language model using the technique mentioned in Section 2.2 to create a set of mixture adapted language models pertaining to every perplexity value in the range. These language models were then used in a PBSMT model where the translation model was trained just on the parallel TED corpora, and tested against the devset. The model which provided the best BLEU scores in the range was selected as the final adapted language model to be used in all further stages of experiments. We used a simple untuned PBSMT model (component weights not set using MERT) for this set of experiments under the assumption that the language model providing the best score in an untuned setting would provide the best score when tuned using MERT.

Notably, this adaptation was only restricted to language models using only the target side (En) of the Multi-UN dataset. Experiments involving the use of Multi-UN bilingual data to enhance the translation models actually resulted in lower scores than the baseline model. The major reason behind this could be attributed to the difference in style between the 'in-domain' and 'out-of-domain' training corpus which affected the phrase-alignments learnt on the 'in-domain' data.

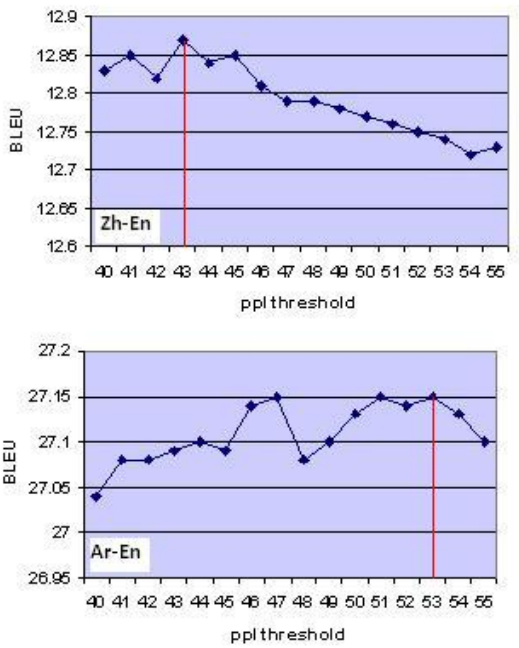

Figure 1: *Perplexity-based threshold values v/s translation quality in BLEU for optimal threshold selection in PBSMT experiments*

Figure 1 shows the variation of BLEU scores for different adapted language models pertaining to different thresholds. According to our experiments, the best cut-off thresholds were 43.00 and 53.00 for Zh–En and Ar–En language pairs, respectively. For Ar–En language pair, the best BLEU

[5] http://www1.ccls.columbia.edu/ cadim/MADA.html

score is achieved for multiple thresholds, and we select the one with the maximum number of sentences in it. The number of Multi-UN sentences thus selected were 55,841 and 89,310 for Zh–En and Ar–En language pairs, respectively.

3.4. HPB Experiments

We built our HPB baseline using the Moses Chart Decoder [24]. Continuous phrases are extracted according to the phrase based system settings explained in Section 3.1. Maximum phrase length and maximum rule span are both set to 12 words. The maximum span for the chart during decoding is set to 20 words, above which only monotone concatenation of phrases is used. Rules extracted contain up to 2 non-terminals. Adjacent non-terminals on the source side are not allowed.

3.5. CCG-augmented HPB Experiments

We built our CCG-augmented HPB system using the Moses Chart Decoder, which has an option to extract syntax-augmented rules from an annotated corpus. We used the same rule extraction and decoding settings as for the HPB baseline system. We used CCG parser and supertagger from the C&C tools [6] to parse the training data for our CCG-augmented HPB systems. We built four CCG-augmented HPB systems using the labeling methods explained in Section 2.4.3:

- HPB-CCG: uses whole CCG categories as non-terminal labels.
- HPB-CCG context: uses CCG contextual labels as non-terminal labels.
- HPB-CCG (s): uses feature-dropped CCG labels as non-terminal labels.
- HPB-CCG context (s): uses feature-dropped CCG contextual labels as non-terminal labels.

4. Experimental Results

This section reports the results for the different set of experiments on Ar–En and Zh–En datasets using TED data and mixture adaptation of language models.

4.1. Arabic–English Translation Results

Table 2 shows BLEU, TER and METEOR scores for the baseline and CCG-based HPB systems on Ar–En translation using just TED data for the translation and language models. HPB-CCG contextual labels system was the best performing system in terms of BLEU, outperforming the PB and HPB baseline systems by 0.1 and 0.12 absolute BLEU points (0.42% and 0.51% relative), respectively. However, these improvements are not statistically significant [25]. The results also show that dropping features from the CCG categories and contextual labels had a negative effect on performance.

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

Table 2: *Experiment results for Ar–En translation using in-domain TED data only.*

System	BLEU	METEOR	TER
PB	23.47	**53.91**	57.77
HPB	23.45	53.57	57.37
HPB-CCG	23.36	53.52	**57.25**
HPB-CCG (s)	23.32	53.52	57.74
HPB-CCG context	**23.57**	53.80	57.27
HPB-CCG context (s)	23.12	53.60	58.30

Table 3 shows the evaluation results for the baseline and CCG-based HPB systems on Ar–En translation using TED data for the translation model and mixture adapted language models. Using mixture adaptation of language model leads to an increase of 5.99 absolute BLEU points (25.41% relative) for the best performing system (CCG contextual labels system) over the corresponding TED-trained model score in Table 2. Language model adaptation also caused the PB-SMT model scores to improve by 5.16 absolute BLEU points (21.99% relative) over the corresponding unadapted PBSMT models. As with the unadapted systems, the HPB-CCG contextual labels system is also the best performing system within all the systems with adapted language models, across all evaluation metrics. It outperformed the mixture-model adapted HPB systems by a statistically insignificant 0.1 absolute BLEU points (0.34% relative). However, it improved over the UN-enhanced mixture-model adapted PB system by 0.93 absolute BLEU points (3.25% relative) providing a statistically significance at p-level=0.05. The results further demonstrate that dropping features from CCG labels caused the performance of the CCG-based systems to deteriorate.

Table 3: *Experiment results for Ar–En translation using mixture adaptation of language models.*

System	BLEU	METEOR	TER
PB	28.63	56.01	55.11
HPB	29.46	56.72	55.64
HPB-CCG	29.22	57.41	55.40
HPB-CCG (s)	28.79	56.57	55.75
HPB-CCG context	**29.56**	**57.63**	**54.89**
HPB-CCG context (s)	29.30	57.19	55.29

For the Ar–En translation task, the best performing system i.e. the HPB-CCG contextual labels system (HPB-CCG context) was submitted as the primary run in the evaluation campaign.

4.2. Chinese-English Translation Results

Table 4 shows the evaluation scores for the baseline and CCG-based HPB systems for Zh–En translation using only TED data for the translation and language models. The results show that different HPB-based systems performed more-or-less similarly, all out-performing the baseline PB system. The feature-dropped CCG labels system was the best

preforming system, beating the HPB baseline system by a small margin of 0.05 absolute BLEU points and also outperforming the PBSMT baseline system by 1.53 absolute BLEU points (14.39% relative) which was statistically significant at p-level=0.05. Notably, dropping features from CCG categories improved the performance of the CCG-based HPB system, while the same had a negative effect on the performance of the HPB-CCG contextual labels system.

Table 4: *Experiment results for Zh–En translation using in-domain TED data only*

System	BLEU	METEOR	TER
PB	10.63	32.68	79.32
HPB	12.11	35.46	76.22
HPB-CCG	12.00	35.53	75.73
HPB-CCG (s)	**12.16**	35.42	**75.36**
HPB-CCG context	12.12	**35.88**	75.87
HPB-CCG context (s)	12.03	35.09	76.46

Table 5 demonstrates the evaluation results for the Zh–En PBSMT, HPB and CCG-augmented HPB systems using TED data for the translation model and mixture adaptation for the language models. Mixture adapted language models allowed the PBSMT model to improve by a score of 1.25 absolute BLEU points (11.76% relative) over the unadapted PBSMT models. Although statistically significant, this improvement was much smaller compared to corresponding improvement noticed for the Ar-En language pairs in Section 4.1. One major reason for this variation could be the huge difference in the size of the additional 'out-of-domain' Multi-UN data used for adaptation between the two language pairs. As pointed out in Section 3.3, Zh–En language pair had 33,829 lesser sentences than the Ar–En language pair for adaptation.

Table 5: *Experiment results for Zh–En translation using mixture adaptation of language models.*

System	BLEU	METEOR	TER
PB	11.88	33.83	85.48
HPB	12.28	**34.57**	**82.85**
HPB-CCG	12.15	34.07	83.18
HPB-CCG (s)	11.47	33.47	84.68
HPB-CCG context	11.94	34.20	83.77
HPB-CCG context (s)	**12.32**	33.89	83.56

The feature-dropped CCG contextual labels system was the best performing system outperforming the HPB and PB mixture-model baseline systems by 0.04 absolute BLEU points (0.33% relative) and 0.44 absolute BLEU points (3.7% relative), respectively. Although the improvement over HPB mixture-model is not statistically significant, that over the PB system is statistically significant at p-level=0.05. The results also show that mixture adaptation of language models improved the performance the best performing system, namely the HPB-CCG contextual labels system by 0.16 absolute BLEU points (1.33% relative) over the best scores for unadapted models in Table 4. As for Ar–En, the best performing system, which is feature-dropped CCG contextual labels system (HPB-CCG Context(s)) was submitted as the primary run for Zh–En translation task.

5. Conclusion

We provide a description of the MT systems built for our participation in the Ar–En and Zh–En MT track as a part of the IWSLT-2011 Evaluation Campaign. We used mixture adaptation of in-domain and out-of-domain language models as an adaptation technique that provided significant improvements over the baseline models built only on in-domain data. We also incorporated CCG into the target side of the HPB SMT system by attaching CCG-extracted labels to non-terminals in hierarchical rules. We tested several CCG-based labelling approaches which examined different complexity levels of non-terminal labels by reducing the amount of information represented in them in order to form a balance between label accuracy and sparsity. Our experiments also showed that mixture adapted language models paired with CCG-based non-terminal labels achieved the best performance for both language pairs. Furthermore, the experiments demonstrated that different CCG-based systems benefited from language model adaptation to different degrees. Lastly, simplifying CCG non-terminal labels helped to improve the score in some cases, while it worsened the performance in the others. The behaviour of different CCG-based labels seems to be affected by the size of the language model and the language pair.

6. Acknowledgments

This work is supported by Science Foundation Ireland (Grant No. 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Dublin City University.

7. References

[1] M. Federico, L. Bentivogli, M. Paul, and S. Stueker, "Overview of the iwslt 2011 evaluation campaign," in *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, 2011.

[2] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *HLT-NAACL 2003: conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series*, Edmonton, Canada, 2003, pp. 48–54.

[3] D. Chiang, "A hierarchical phrase-based model for statistical machine translation," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ser. ACL '05, 2005, pp. 263–270.

[4] M. Federico and N. Bertoldi, "Broadcast news lm adaptation using contemporary texts," in *Proceedings of*

European Conference on Speech Communication and Technology (Eurospeech), 2001, pp. 239–242.

[5] M. Steedman, *The syntactic process*. Cambridge, MA, USA: MIT Press, 2000.

[6] 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, and E. Herbst, "Moses: open source toolkit for statistical machine translation," in *ACL 2007, Proceedings of the Interactive Poster and Demonstration Sessions*, Prague, Czech Republic, 2007, pp. 177–180.

[7] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, Sapporo Convention Center, Japan, 2003, pp. 160–167.

[8] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. NY, USA: Springer New York Inc., 2001.

[9] P. Banerjee, S. K. Naskar, J. Roturier, A. Way, and J. Van Genabith, "Domain adaptation in statistical machine translation of user-forum data using component-level mixture modelling," in *Proceedings of the 13th Machine Translation Summit*, 2011, pp. 285–292.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, B*, vol. 39, pp. 1–38, 1977.

[11] A. Zollmann and A. Venugopal, "Syntax augmented machine translation via chart parsing," in *Proceedings of the Workshop on Statistical Machine Translation*, ser. StatMT '06, 2006, pp. 138–141.

[12] H. Almaghout, J. Jiang, and A. Way, "CCG augmented hierarchical phrase-based machine translation," in *Proceedings of the 7th International Workshop on Spoken Language Translation*, Paris, France, December 2010, pp. 211 – 218.

[13] S. Bangalore and A. Joshi, "Supertagging: An approach to almost parsing," *Computational Linguistics*, vol. 25, no. 2, pp. 237–265, 1999.

[14] H. Almaghout, J. Jiang, and A. Way, "CCG contextual labels in hierarchical phrase-based SMT," in *proceedings of the 15th conference of the European Association for Machine Translation*, Leuven, Belgium, 2011, pp. 281–288.

[15] N. Stroppa and A. Way, "MATREX: DCU Machine Translation System for IWSLT 2006," in *IWSLT 2006: Proceedings of the 3rd International Workshop on Spoken Language Translation*, Palulu Plaza, Kyoto, Japan, 2006, pp. 31–36.

[16] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, pp. 19–51, 2003.

[17] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, (ACL 2002)*, Philadelphia, Pennsylvania, 2002, pp. 311–318.

[18] M. Federico, N. Bertoldi, and M. Cettolo, "IRSTLM: an open source toolkit for handling large scale language models," in *Interspeech 2008: 9th Annual Conference of the International Speech Communication Association*, Brisbane, Australia, 2008, pp. 1618–1621.

[19] R. Kneser and V. Steinbiss, "On the dynamic adaptation of stochastic language models," in *Proceedings of the 1993 IEEE international conference on Acoustics, speech, and signal processing: speech processing - Volume II*, ser. ICASSP'93, Minneapolis, Minnesota, USA, 1993, pp. 586–589.

[20] A. Lavie and A. Agarwal, "Meteor: an automatic metric for mt evaluation with high levels of correlation with human judgments," in *Proceedings of the Second Workshop on Statistical Machine Translation*, 2007, pp. 228–231.

[21] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *In Proceedings of Association for Machine Translation in the Americas*, 2006, pp. 223–231.

[22] H. Tseng, P. Chang, G. Andrew, D. Jurafsky, and C. Manning, "A conditional random field word segmenter," in *In Fourth SIGHAN Workshop on Chinese Language Processing*, 2005.

[23] R. Haque, S. K. Naskar, J. Van Genabith, and A. Way, "Experiments on Domain Adaptation for English–Hindi SMT," in *Proceedings of PACLIC 23: the 23rd Pacific Asia Conference on Language, Information and Computation*, Hong Kong, 2009, pp. 670–677.

[24] H. Hoang, P. Koehn, and A. Lopez, "A Unified Framework for Phrase-Based, Hierarchical, and Syntax-Based Statistical Machine Translation," in *Proc. of the International Workshop on Spoken Language Translation*, Tokyo, Japan, 2009, pp. 152–159.

[25] P. Koehn, "Statistical Significance Tests for Machine Translation Evaluation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing, (EMNLP 2004)*, Barcelona, Spain, 2004, pp. 388–395.