Low-Resource Machine Translation Using MATREX: The DCU Machine Translation System for IWSLT 2009

Yanjun Ma, Tsuyoshi Okita, Özlem Çetinoğlu, Jinhua Du, Andy Way

Centre for Next Generation Localisation
School of Computing
Dublin City University
Dublin 9, Ireland
{yma,tokita,ocetinoglu,jdu,away}@computing.dcu.ie

Abstract

In this paper, we give a description of the Machine Translation (MT) system developed at DCU that was used for our fourth participation in the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT 2009). Two techniques are deployed in our system in order to improve the translation quality in a low-resource scenario. The first technique is to use multiple segmentations in MT training and to utilise word lattices in decoding stage. The second technique is used to select the optimal training data that can be used to build MT systems. In this year's participation, we use three different prototype SMT systems, and the output from each system are combined using standard system combination method. Our system is the top system for Chinese–English CHALLENGE task in terms of BLEU score.

1. Introduction

In this paper, we describe some new extensions to the hybrid data-driven MT system developed at DCU, MATREX (Machine Translation using Examples), subsequent to our participation at IWSLT 2006 [1], IWSLT 2007 [2] and IWSLT 2008 [3]. In this year's participation, optimising the system in a low-resource scenario is our main focus.

The first technique deployed in our system is word lattice decoding, where the input of the system is not a string of words, but rather a lattice encoding multiple segmentations for a single sentence. This method has been repeatedly demonstrated to be effective in improving the coverage of the MT systems [4, 5, 6, 7]. Another technique investigated is a novel data selection method, which differentiates high- and low-quality bilingual sentence pairs in the training data, and use them separately in training MT systems.

We participate in the CHALLENGE tasks and the BTEC Chinese–English and Turkish–English tasks. For CHALLENGE tasks, both the single-best ASR hypotheses and the correct recognition results are translated. Three different prototype SMT systems are built for each translation task and a few novel techniques are then applied to different systems. The final submission is a combination of the outputs from different systems.

The remainder of the paper is organized as follows. In Section 2, we describe the various components of the system; in particular, we give details about the various novel extensions to MATREX as summarised above. In Section 3, the experimental setup is presented and experimental results obtained for various language pairs are reported in Section 4. In Section 5, we conclude, and provide avenues for further research.

2. The MATREX System

The MATREX system is a hybrid data-driven MT system which exploits aspects of different MT paradigms [1]. The system follows a modular design and facilitates the incorporation of different MT engines and novel techniques. In this year's participation, besides additional MT engines, the system has also been extended with a system combination module which can combine different MT outputs [8]. In the following subsections, we describe the main techniques used in the participation of IWSLT 2009.

2.1. Word Lattice

To mitigate the negative effects of the inaccurate word segmentation, word lattice, which encodes a few alternative segmentations for a given sentence, can be used as input [5, 6] of the MT systems. This technique can be applied to languages where the word boundaries are not orthographically marked such as Chinese, or languages with rich morphology such as Turkish.

In the decoding stage, the various segmentation alternatives can be encoded into a compact representation of word lattices. A **word lattice** $G = \langle V, E \rangle$ is a directed acyclic graph that formally is a weighted finite state automaton. In the case of word segmentation, each edge is a candidate word associated with its weights. A straightforward estimation of the weights is to distribute the probability mass for each node

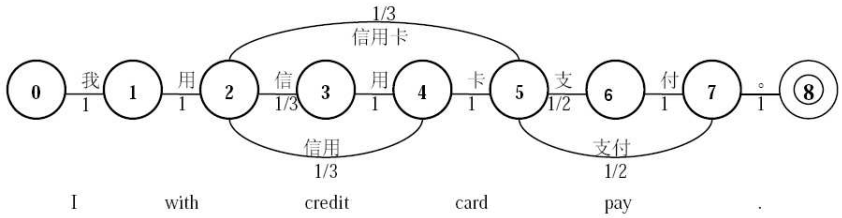

Figure 1: An example of a word lattice for a Chinese sentence

uniformly to each outgoing edge.[1] The single node having no outgoing edges is designated as the "end node". An example of a word lattice for a Chinese sentence is shown in Figure 1.

2.1.1. Word Lattice Generation

Multiple segmenters are used to segment the Chinese and Turkish sentences and the segmentation results are combined into a word lattice. For Chinese, we used the original manual segmentation, the LDC segmentation obtained from the LDC segmenter and character-based segmentation simply by splitting a sentence into characters. For Turkish, we used two segmentations. The first segmentation uses the lowercased original data, i.e., each word is a segment. For the second segmentation, we morphologically analysed and disambiguated the data [9]. Then we applied a selective approach where only the informative morphemes are kept in the morphological representation. For instance the pronoun *bana* 'to me' has the morphological analysis `ben+Pron+Pers+A1sg+Pnon+Dat`.[2] In the selective approach, it is reduced to `ben+Pron+Dat` Finally, this reduced representation is splitted into morphemes so that each morpheme corresponds to a segment.

2.1.2. Phrase-Based Word Lattice Decoding

Given a Chinese input sentence f_1^I consisting of I characters, the traditional approach is to first determine the best word segmentation and perform decoding afterwards. In such a case, we first seek a single best segmentation, as in (1):

$$\hat{v}_1^K = \arg\max_{v_1^K,K}\{P(v_1^K|f_1^I)\} \quad (1)$$

Then in the decoding stage, we seek the translation of the most likely source segmentation, as in (2):

$$\hat{e}_1^J = \arg\max_{e_1^J,J}\{P(e_1^J|\hat{v}_1^K)\} \quad (2)$$

In such a scenario, some segmentations which are potentially optimal for translation may be lost. This motivates the need for word lattice decoding. The decision rules (1) and (2) can be rewritten as in (3)–(5):

$$\hat{e}_1^J = \arg\max_{e_1^J,J}\{\max_{v_1^K,K} P(e_1^J, v_1^K|f_1^I)\} \quad (3)$$

$$= \arg\max_{e_1^J,J}\{\max_{v_1^K,K} P(e_1^J)P(v_1^K|e_1^J, f_1^I)\} \quad (4)$$

$$\simeq \arg\max_{e_1^J,J}\{\max_{v_1^K,K} p(e_1^J)p(v_1^K|f_1^I)p(v_1^K|e_1^J)\} \quad (5)$$

where $p(e_1^J)$ is the language model, $p(v_1^K|f_1^I)$ is the word segmentation model and $p(v_1^K|e_1^J)$ is the translation model. Compared to the decision rule of the standard source-channel model for Statistical Machine Translation (SMT), (5) has an additional segmentation model.

Given the fact that the number of segmentations K grows exponentially with respect to the number of characters J, it is impractical to firstly enumerate all possible v_1^K and then to decode. However, it is possible to enumerate all the alternative segmentations for a substring of f_1^I which contains a very limited number of characters, making the utilisation of word lattices tractable in Phrase-Based SMT (PB-SMT).

2.2. Data Selection Techniques

Given that the amount of training data available for these tasks are limited, developing techniques to make the best use of them is essential to the performance of the MT systems. We used two techniques to improve the translation model by differentiating "good" and "bad" data.

The first technique, namely *good points algorithm*, selects high-quality parallel sentence pairs in the training data to build MT systems. This leads to better word alignments since this process can remove noisy sentence pairs (also called outliers) from training data. Given that state-of-the-art word alignment models only allows 1-to-n mappings between source and target words, those sentences which include n-to-m mappings between source and target words (for example, paraphrases, non-literal translations, and multi-word expressions) are considered to be noise. The noisy sentence pairs can potentially hinder a word aligner in achieving high quality alignments; moreover, the errors in word alignment will be propagated in later stages of MT training including phrase extraction. To remove the noisy sentence pairs, we use a method as shown in Algorithm 1 [10].

[1] We can also use language models to assign probabilities to each edge as in [4]. In this case, however, we have to rely on some segmented data to train the language model.

[2] +Pron: pronoun, +Pers: personal, +A1sg: 1st person singular, +Pnon: no possessive +Dat: dative

Algorithm 1 Good Points Algorithm

Step 1: Train word-based SMT using the whole training data, and translate all the sentences in the training data to output n-best lists.
Step 2: For the n-best translations for each source sentence, obtain the (maximum) cumulative X-gram ($X \in \{1, \cdots, 4\}$) score $S_{WB,X}$ by comparing each translation against the reference target sentence. This score is used measure the quality of the current sentence pair.
Step 3: Train PB-SMT using the whole training data. Translate all training sentences to output n-best lists.
Step 4: For the n-best translations for each source sentence, obtain the (maximum) cumulative X-gram ($X \in \{1, \cdots, 4\}$) score $S_{PB,X}$ by comparing each translation against the reference target sentence. This score is also used measure the quality of the current sentence pair.
Step 5: Remove sentence pairs where $S_{WB,2} = 0$ and $S_{PB,2} = 0$, and sentence length is greater than 2.
Step 6: The remaining sentence pairs after removal in Step 5 are used to train the final PB-SMT systems.

Different from translations between European languages (e.g. from Spanish, German and French to English) where outliers were around 5 percent, we obtained around 10 percent outliers for Chinese–English translation task, and 3 percent for Turkish–English. We observed that word alignment becomes worse if more than 10 percent of the sentence pairs are treated as outliers and removed. Hence, our algorithm requires SMT to output an n-best list in translating each source sentence, and score each candidate in the n-best list. The maximum score obtained is used to score current sentence pair. Some of the Chinese–English sentence pairs detected as outliers are shown in Table 1.

总共 是 多少 ? what does that come to ?
服务台 的 号码 是 多少 ? what number should i dial for information ?
它 在 星期几 开 ? what days of the week does it take place ?
这 是 钥匙 。 the keys go here .
一 点 过 五 分 。 it 's five after one .

Table 1: Outliers for BTEC Chinese–English task by Good Point algorithm.

2.3. Multiple System Combination

Multiple system combination technique [8] is deployed to combine the outputs from three different prototype Statistical Machine Translation systems, namely PB-SMT, Hierarchical Phrase-Based SMT (HPB) and Syntax-Based SMT (SBMT).

For multiple system combination, we implement an Minimum Bayes-Risk-Confusion Network (MBR-CN) framework as used in [8]. Due to the varying word order in the MT hypotheses, it is essential to decide the backbone which determines the general word order of the confusion network. Instead of using a single system output as the skeleton, we employ a MBR decoder to select the best single system output from the merged N-best list by minimising the BLEU [11] loss, as in (6):

$$\hat{e}_i = \underset{i \in \{1,\cdots,N\}}{\arg\min} \sum_{j=1}^{N} \{1 - BLEU(e_j, e_i)\} \quad (6)$$

where e_i and e_j are hypotheses in the N-best list, and N indicates the number of hypotheses in the merged N-best list. $BLEU(e_j, e_i)$ calculates sentence-level BLEU score of e_i with e_j as the reference translation.

The confusion network is built using the output of MBR decoder as the backbone which determines the word order of the combination. The other hypotheses are aligned against the backbone based on the TER metric. NULL words are allowed in the alignment. Either votes or some form of confidence measures are assigned to each word in the network. Each arc in the CN represents an alternative word at that position in the sentence and the number of votes for each word is counted when constructing the network. The features we used are as follows:

- word posterior probability [12]
- trigram and 4-gram target language model
- word length penalty
- NULL word length penalty

Minimum Error-Rate Training (MERT) is used to tune the weights of the confusion network.

2.4. Case and Punctuation Restoration

Given that the English data are lower cased in MT training, the restoration of the case information is required for both BTEC and CHALLENGE tasks. For CHALLENGE tasks where the input is speech recognition results, punctuation restoration is also required. In order to obtain better word alignments for our MT system, we trained our system on data with punctuation. Therefore, punctuation restoration is performed as a preprocessing step preceding translation.

For punctuation restoration, it is possible to consider punctuation marks as hidden events occurring between words, with the most likely hidden tag sequence (consistent with the given word sequence) being found using an *n*-gram language model trained on a punctuated text. For case restoration, the task can be viewed as a disambiguation task in which we have to choose between the (case) variants of

each word of a sentence. Again, finding the most likely sequence can be done using an *n*-gram language model trained on a case-sensitive text.

We used a translation-based approach [2] treating case restoration as a translation task, where the lower-cased sentences are the "source" language and the true-cased sentences are the "target". Regarding punctuation restoration, the text with punctuation can be considered as the target language. Then we remove the punctuation in the target language and use them as the corresponding source language to construct a pseudo-'bilingual' corpus. With this 'bilingual' corpus, we can train a phrase-based SMT system to restore punctuation. Naturally we can also train a system to restore case information only, or if required, to restore both case information and punctuation.

We observed that the final punctuation mark is the most difficult to be restored. The language model(LM)-based approach can propose two conflicting hypotheses, while the translation-based approach suffers from translation quality. In order to better restore the final punctuation mark, we combine the output of LM and translation-based approaches with a majority voting procedure. With two proposed hypotheses from the LM-based method and one from the translation-based method, we choose the hypothesis using majority voting. If no solution can be found using this approach, we choose the first hypothesis proposed by the LM-based method.

3. Experimental Setup

In our experiments, we used data provided within the evaluation campaign; no additional data resources are used. The detailed data setting will be explained when we report the experimental results for each task. In addition to the original manual segmentation, we used LDC segmenter to segment Chinese sentences. In order to train Syntax-Based SMT systems, we need to parse the sentences in target language, i.e. Chinese or English in our case. Berkeley parser [13] with default setting is used to parse both Chinese and English sentences.

The GIZA++ implementation [14] of IBM Model 4 [15] is used as the baseline for word alignment, and the "Grow-Diag-Final" (GDF) and intersection (INT) heuristics[3] described in [16] to derive the refined alignment from bidirectional alignments. Model 4 is incrementally trained by performing 5 iterations of Model 1, 5 iterations of HMM, 3 iterations of Model 3, and 3 iterations of Model 4.

The baseline in our experiments is a standard log-linear PB-SMT system. With the word alignment obtained using the above-mentioned method, we perform phrase-extraction using heuristics described in [16], MERT [17] optimising the BLEU metric, a 5-gram language model with Kneser-Ney smoothing [18] trained with SRILM[4] [19] on the English side of the training data, and MOSES for decoding.

Three open-source SMT systems, i.e. PB-SMT system Moses [20], Hierarchical Phrase-Based system Joshua [21] and Syntax-Based SMT system SAMT [22] are used in our experiments.

4. Experimental Results

In the following subsections, we report some preliminary experimental results we obtained using three different systems (PB-SMT, HPB and SBMT) and the techniques we described above, namely word lattice decoding, data selection algorithm (DS). System combination results (Sys Combo) can be finally obtained based on these single systems.[5]

4.1. BTEC Chinese–English

For BTEC Chinese–English translation task, we used devset7 for development purposes, the rest of the development sets are merged into the training data in our final system.

Table 2 shows the experimental results for this task. For this particular task, PB-SMT enhanced with GDF word alignment heuristics and word lattice decoding achieved the highest performance. HPB and SBMT systems underperform the PB-SMT systems, indicating that syntax does not benefit much for spoken language translation where sentences tend to be short and the parsers trained on news data do not perform well. System combination technique boots the system performance over the best single system (Lattice-GDF) by 1.95 absolute BLEU points, which correspond to a 4.87% relative improvement. We observed gains using data selection method during internal testing of our system. However, in the evaluation campaign, this method does not seem to benefit.

From the amount of OOV words in the translation output, we can see that one of the major advantages of using lattices in such a low-resource scenario is the higher coverage, i.e. smaller number of OOV words. We can also see that using INT heuristic instead of GDF can also improve the coverage because INT heuristic induces fewer word alignment links and more phrases pairs can be extracted based on the word alignment. The OOV words from the system using DS algorithm is higher than others because some sentence pairs are removed from the training data and the coverage is lower.

The restoration of case and punctuation information leads to an increase in BLEU score demonstrating the strength of our case and punctuation restoration component.

4.2. BTEC Turkish–English

For BTEC Turkish–English translation task, there are only two development sets. We used devset1 for development purposes, and devset2 was merged into the training data in

[3] In our experiments, we only tried these two heuristics due to limited amount of time; however, other heuristics are also worth exploiting.

[4] Specifically, we used SRILM release 1.4.6.

[5] Please note that in our primary submission, i.e. the system combination results, the out-of-vocabulary (OOV) words was removed. Therefore, the system combination results reported in this paper can be slightly lower than the official scores.

	PB-SMT			Lattice				
	GDF	INT	DS-GDF	GDF	INT	HPB	SBMT	Sys Combo
case+punc	0.3903	0.3856	0.3733	**0.4002**	0.3672	0.3783	0.3612	**0.4197**
no_case+no_punc	0.3808	0.3717	0.3617	0.3811	0.3463	0.3614	0.3466	0.4135
OOV	139	90	191	40	6	139	141	48

Table 2: Performance of single systems and multiple system combination for BTEC Chinese–English translation (BLEU)

our final system. We used the first segmentation to build a standard PB-SMT system. The second segmentation is used together with the first one to generate word lattices.

Table 3 shows the experimental results for BTEC Turkish–English translation task. Similar to the BTEC Chinese–English task, the PB-SMT system with word lattice decoding achieved better performance than other systems. Specifically the word lattice system with intersection (INT) heuristic for word alignment received the best single system BLEU score. Compared to the BTEC Chinese–English systems, the gains from using lattice in PB-SMT is greater. This is largely due to the fact that Turkish is a morphologically rich language and lattice-based method can substantially improves the coverage. System combination further improves the performance by 3.46 absolute BLEU points over the best single system, corresponding to a 6.59% relative improvement.

HPB and SBMT systems do not show higher performance over PB-SMT systems, due to the same reason we analysed for BTEC Chinese–English translation. The data selection algorithm does not benefit showing the high quality of the training data. Similar phenomena as BTEC Chinese–English system are observed w.r.t OOV words. The restoration of case and punctuation information also contributes the high performance of the systems.

4.3. CHALLENGE Chinese–English

For CHALLENGE Chinese–English translation task, we used devset4 for development purposes, the rest of the development sets (corrected recognition results) were merged into the training data in our final system.

Table 4 shows the experimental results for CHALLENGE Chinese–English translation task. For this task, we observed similar trends in system performance as the BTEC Chinese–English task. Again, lattice-based systems outperform other systems and the system combination results can gain further over the single systems. These gains can be partly explained by the low number of OOV words. Our system combination translation results for corrected speech recognition (CRR), which was submitted as primary system in the evaluation, received the top BLEU score out of all the participants in this task. Our system for translating ASR input is also the top system in translating the single best ASR results.

From Table 4, we also observe that the system performance for ASR translation is much lower than CRR. This indicates the necessity of adapting systems to translate the "imperfect" source texts.

4.4. CHALLENGE English–Chinese

For CHALLENGE English–Chinese translation task, we used devset12 for development purposes, the rest of the development sets (corrected recognition results) were merged into the training data in our final system.

Table 5 exhibits the experimental results for CHALLENGE English–Chinese translation task. Given that English is not a morphologically rich language, lattice-based techniques are not applied. The best performance for CRR translation is achieved by the PB-SMT system with intersection heuristic for word alignment (lowest number of OOV words). For ASR (single best) translation, HPB achieved the highest performance.

System combination does not lead to gains in system performance. This can be attributed to the inconsistency in the development and test data. Figure 2 is the graph showing the performance of our systems on development set (devset) and test set (testset). As can be seen from the graph that on devset and testset there is major difference in the relative order of performance for our five single systems to be combined. On the devset the best system is PB-SMT with intersection (INT) heuristic for word alignment and the worst system is the HPB system. Conversely, on the test set, it turns out that HPB system is the best system and PB-SMT with GDF heuristic is the worst. Such a discrepancy between devset and testset imposes a major challenge for system combination. A better selection of development set is needed in order to make system combination more useful.

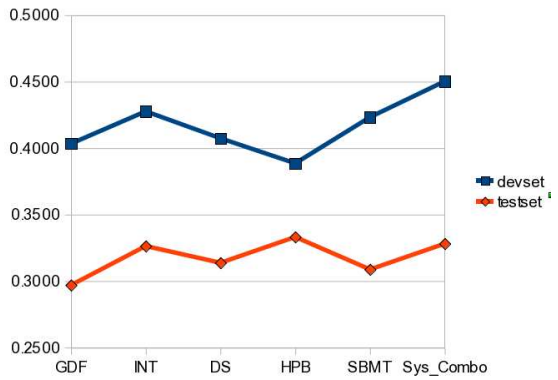

Figure 2: Performance of the systems on development set and test set

	PB-SMT			Lattice				
	GDF	INT	DS-GDF	GDF	INT	HPB	SBMT	Sys Combo
case+punc	0.4831	0.4656	0.4591	0.5233	**0.5247**	0.4711	0.4708	**0.5593**
no_case+no_punc	0.4590	0.4394	0.4390	0.5008	0.5065	0.4455	0.4516	0.5401
OOV	106	61	106	21	11	88	80	17

Table 3: Performance of single systems and multiple system combination for BTEC Turkish–English translation (BLEU)

		PB-SMT			Lattice				
		GDF	INT	DS-GDF	GDF	INT	HPB	SBMT	Sys Combo
CRR	case+punc	0.3169	0.3278	0.3143	**0.3436**	0.3335	0.3148	0.2978	**0.3689**
	no_case+no_punc	0.3109	0.3262	0.3088	0.3371	0.3310	0.3057	0.2906	0.3673
	OOV	197	76	188	21	0	191	197	16
ASR.1	case+punc	0.2918	0.2915	0.2913	0.2724	**0.2958**	0.2869	0.2700	**0.3161**
	no_case+no_punc	0.2789	0.2825	0.2752	0.2660	0.2861	0.2744	0.2536	0.3064
	OOV	158	96	153	5	5	157	154	5

Table 4: Performance of single systems and multiple system combination for CHALLENGE Chinese–English translation (BLEU)

4.5. Further Analysis of the Data Selection Method

Given the fact that data selection method does not work well for our current tasks, we provide some further analysis in a bid to reveal the reasons behind this. Firstly, the percentage of sentences that we removed using our method amounts to 10 to 13 percents. For our experiments for European language pairs this figure was 3 to 5 percents. The sharp decrease of amount of training data can result in lower word alignment quality and the phrase extraction may also be affected. When the removed sentence pairs only amount to 3 to 5 percents, we could improve the quality of training data by removing the noisy sentence pairs.

Secondly, although we could perform a DS algorithm depending on HMM alignment and n-best lists of word-based translation system, we did only a basic procedure due to time limitations. In our preparation phase, we observed that HMM alignment led to a better BLEU score. For a language pair as different as Chinese and English, a lexical translation probability tends to have high entropy. One way to mitigate the negative effects of such high entropies would be to employ n-best lists in the word-based translation system. However, we did not employ this strategy in here due to time limitation.

Another challenge of applying our approach to Chinese and Turkish translation is the word segmentation problem. The word segmentation process can introduce noise in the pipeline of the SMT systems. How to handle such error prone word segmentation would be a future work we need to pin down for DS algorithm.

5. Conclusions

In this paper, we described our new techniques deployed in our MATREX system in order to improve the translation quality in a low-resource scenario. The first technique is to use multiple segmentations in MT training and to utilise word lattices in decoding stage. A second technique is used to select the optimal training data that can can be used to build MT systems. We show that word lattices are useful in such low-resource scenarios. The lattice-based system is our best single system for Chinese–English and Turkish–English translation. The lattice-based method shows greater benefit for Turkish–English translation than Chinese–English further demonstrating its effectiveness in dealing with morphologically rich languages. Our primitive method for data selection does not benefit much in current tasks due to the high quality of the IWSLT training data.

System combination techniques can boot the system performance given a proper development process. For Chinese–English and Turkish–English translations, the best performance is achieved by the system combination. For the CHALLENGE Chinese–English translation task, our system achieved the top BLEU score among systems from different sites worldwide. For English–Chinese translation, we found out that the major discrepancy between the devset and testset resulted in the inferior performance of system combination.

6. Acknowledgements

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (`http://www.cngl.ie`) at Dublin City University. We would like to thank the Irish Centre for High-End Computing.[6] We would also like to thank Kemal Oflazer for providing us the morphological analyser output and the selective segmentation data.

7. References

[1] N. Stroppa and A. Way, "MaTrEx: the DCU Machine Translation system for IWSLT 2006," in *Proceedings*

[6] `http://www.ichec.ie/`

		PB-SMT					
		GDF	INT	DS-GDF	HPB	SBMT	Sys Combo
CRR	case+punc	0.3531	0.3833	0.3547	0.3797	0.3563	0.3725
	no_case+no_punc	0.3555	**0.3885**	0.3570	0.3832	0.3613	**0.3757**
	OOV	99	32	91	102	101	38
ASR.1	case+punc	0.2970	0.3264	0.3138	0.3332	0.3088	0.3273
	no_case+no_punc	0.2987	0.3315	0.3154	**0.3372**	0.3110	**0.3306**
	OOV	129	64	141	112	120	40

Table 5: Performance of single systems and multiple system combination for BTEC English–Chinese translation (BLEU)

of the International Workshop on Spoken Language Translation, Kyoto, Japan, 2006, pp. 31–36.

[2] H. Hassan, Y. Ma, and A. Way, "MaTrEx: the DCU Machine Translation system for IWSLT 2007," in *Proceedings of the International Workshop on Spoken Language Translation*, Trento, Italy, 2007, pp. 21–28.

[3] Y. Ma, J. Tinsley, H. Hassan, J. Du, and A. Way, "Exploiting alignment techniques in MaTrEx: the DCU Machine Translation system for IWSLT08," in *Proceedings of International Workshop on Spoken Language Translation (IWSLT08)*, Honolulu, HI, 2008, pp. 26–33.

[4] J. Xu, E. Matusov, R. Zens, and H. Ney, "Integrated Chinese word segmentation in Statistical Machine Translation," in *Proceedings of the International Workshop on Spoken Language Translation*, Pittsburgh, PA, 2005, pp. 141–147.

[5] C. Dyer, S. Muresan, and P. Resnik, "Generalizing word lattice translation," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Columbus, OH, 2008, pp. 1012–1020.

[6] Y. Ma and A. Way, "Bilingually motivated domain-adapted word segmentation for Statistical Machine Translation," in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009)*, Athens, Greece, 2009, pp. 549–557.

[7] ——, "Bilingually motivated word segmentation for Statistical Machine Translation," *ACM Transactions on Asian Language Information Processing, Special Issue on Machine Translation of Asian Languages*, vol. 8, no. 2, pp. 1–24, 2009.

[8] J. Du, Y. He, S. Penkale, and A. Way, "MaTrEx: The DCU MT system for WMT 2009," in *Proceedings of the Fourth Workshop on Statistical Machine Translation*, Athens, Greece, 2009, pp. 95–99.

[9] H. Sak, T. Güngör, and M. Saraçlar, "Morphological disambiguation of Turkish text with perceptron algorithm," in *CICLing 2007*, vol. LNCS 4394, 2007, pp. 107–118.

[10] T. Okita, "Data cleaning for word alignment," in *ACL 2009 Student Research Workshop*, Singapore, 2009, pp. 72–80.

[11] 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*, Philadelphia, PA, 2002, pp. 311–318.

[12] J. G. Fiscus, "A post-processing system to yield reduced Word Error Rates: Recogniser output voting error reduction (ROVER)," in *Proceedings 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, Santa Barbara, CA, 1997, pp. 347–352.

[13] S. Petrov, L. Barrett, R. Thibaux, and D. Klein, "Learning accurate, compact, and interpretable tree annotation," in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, 2006, pp. 433–440.

[14] F. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[15] P. F. Brown, S. A. Della-Pietra, V. J. Della-Pietra, and R. L. Mercer, "The mathematics of Statistical Machine Translation: Parameter estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[16] P. Koehn, F. Och, and D. Marcu, "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.

[17] F. Och, "Minimum Error Rate Training in Statistical Machine Translation," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan, 2003, pp. 160–167.

[18] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, Detroit, MI, 1995, pp. 181–184.

[19] A. Stolcke, "SRILM – An extensible language modeling toolkit," in *Proceedings of the International Conference on Spoken Language Processing*, Denver, CO, 2002, pp. 901–904.

[20] 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 *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, Prague, Czech Republic, 2007, pp. 177–180.

[21] Z. Li, C. Callison-Burch, C. Dyer, S. Khudanpur, L. Schwartz, W. Thornton, J. Weese, and O. Zaidan, "Joshua: An open source toolkit for parsing-based machine translation," in *Proceedings of the Fourth Workshop on Statistical Machine Translation*, Athens, Greece, 2009, pp. 135–139.

[22] A. Zollmann and A. Venugopal, "Syntax augmented Machine Translation via chart parsing," in *Proceedings of the Workshop on Statistical Machine Translation*, New York City, NY, 2006, pp. 138–141.