The CNGL MT System for CWMT'2013

Liangyou Li Jian Zhang Chris Hokamp Xiaofeng Wu Xiaojun Zhang Qun Liu
Centre for Next Generation Localisation, School of Computing
Dublin City University,
Dublin 9, Ireland
{liangyouli,zhangj,chokamp,xiaofengwu,xzhang,qliu}@computing.dcu.ie

第九届全国机器翻译研讨会（CWMT'2013）

CNGL 技术报告

李良友，张健，Chris Hokamp，吴晓峰，张霄军，刘群
下一代本地化中心, 计算机学院
都柏林城市大学,
都柏林 9 区, 爱尔兰
{liangyouli,zhangj,chokamp,xiaofengwu,xzhang,qliu}@computing.dcu.ie

Abstract

This paper describes the six machine translation systems we submitted to the 9th Chinese Workshop on Machine Translation (CWMT'2013). We employed different combinations of several techniques, including multiple phrase extraction, sparse features, and operation sequence model (OSM), on these tasks. In this paper, we explain these systems and our experimental results on the development set. We also report on our pre-processing methodology.

摘要

本文介绍了 CNGL 在第九届全国机器翻译研讨会（CWMT'2013）上的参赛系统。这些系统使用了几个技术，包括多对齐、稀疏特征和操作序列模型。同时本文还报告了语料的预处理过程和开发集上实验结果。

1 Introduction

This report describes CNGL of Dublin City University (DCU) submission for CWMT'2013. There are six tasks in this challenge, which are the Chinese-English news task (CE news), English-Chinese news task (EC news), English-Chinese Science &Technology task (EC s&t), Tibetan-Chinese government task (TC), Uyghur-Chinese news task (UC) and Mongolian-Chinese task (MC). We carried out experiments on all of the tasks by using different statistical techniques and translation models.

Two of the widely used statistical machine translation models are the phrase-based model (Koehn et al., 2007) and the hierarchical phrase-based model (Chiang, 2005). In the CWMT'2013 we utilized both models for our experiments.

In recent years, researchers have developed many techniques to boost the performance of those models. We employed three of these techniques in our work: multiple phrase extraction, sparse features and the operation sequence model (OSM) (Durrani et al.,2011; Durrani *et al*.,2013; Schütze, 2013). These techniques are integrated individually, or in combination, into each of the submitted systems either used to improve the baseline performance.

Model	Feature Description	
Hierarchical phrase-based model	Source side	The cluster ID of word immediately to the left of rule
		The cluster ID of word immediately to the right of rule
		The cluster ID of first word of *k*th non-terminal X_k covers
		The cluster ID of last word cluster of *k*th non-terminal X_k covers
Phrase-based model	Both source and target side	The cluster ID of word immediately to the left of phrase
		The cluster ID of word immediately to the right of phrase
		The cluster ID of first word of phrase
		The cluster ID of last word of phrase

Table 1: Sparse features for rule or phrase pair selection

The rest of this paper is organized as follows: Section 2 gives details of the techniques we used in the challenge. In Section 3, we describe the experimental set-up, including the preprocessing of the corpus, and provide our results on the development sets. Section 4 gives our conclusions and suggests some future work.

2 Techniques Description

1.1 Multiple phrase extraction

Multiple phrase extraction aims to exploit complementarity in diverse phrase extraction methods. The phrase extractors are defined on different alignments with the same extraction algorithm. For the training corpus, we use different aligners to train several alignments. Then extraction is preceded on these alignments.

In our systems, $c(s,t)$ is used to estimate translation probabilities $p(s|t)$ and $p(t|s)$.

$$c(s,t) = \sum_{m=1}^{M} c(s,t,m) \quad (1)$$

For each phrase pair (s,t), where *s* refers to source language phrase and *t* to target language phrase, the extraction count from alignment *m* is $c(s,t,m)$.

In this paper, multiple phrase extraction is only performed for the three minority language tasks (TC, MC, and UC), since they have smaller training corpora. For these language pairs, we used two different alignments: the GIZA++ alignment (Och and Ney, 2003) with heuristic function "intersect" and a maximum phrase length of 20, and the alignment from the Berkeley aligner (Liang et al., 2006). Table 1 shows the number of phrase pairs or rules extracted from the different alignments on the three minority language tasks.

Task	GIZA++ (intersect)	Berkeley	multi-extraction
TC	141,847,128	13,493,474	146,219,668
UC	80,192,461	10,720,206	81,986,203
MC	106,664,492	10,791,350	110,463,024

Table 1: the number of rules or phrase pairs extracted from different alignment

1.2 Sparse Features

For each source phrase or rule, there is usually more than one corresponding translation option. Each different translation may be optimal in different contexts. However, the probability distributions estimated by Maximum Likelihood would make the translation prefer the most common translation. Thus in our systems, features which describe the context difference between phrases or rules, are designed to select the right translation according to specific circumstances.

He et al. (2008) proposed a rule selection method for the hierarchical phrase-based model. They trained Maximum Entropy (ME) Model for each ambiguous source side of rule. During decoding, according to features from the input sentence, this ME model makes prediction for each rule. This prediction is viewed as an additional run-time feature added to the log-linear model. However, when the corpus is large, the ME training procedure would be time-consuming and not practical.

Instead, we use sparse features which characterize each rule or phrase pair directly. We designed a different feature set for the two different translation models. For the hierarchical phrase based model, only the source side is considered. Both context information of the rule and the non-terminals in the rule are used to extract features. This is the same with (He et al., 2008). For the phrase based model, we extract features from both source and target side. Table 1 lists the features used in our systems.

Since He et al. (2008) has shown features based on POS are more suitable for rule selection, in our systems, we converted the context features mentioned above into POS-like features. In order to make our sparse features language independent, all features are generalized with cluster ID, because POS taggers for Uygur, Mongolian, and Tibetan were not available. To obtain POS-like features in an unsupervised manner, *mkcls* in GIZA++ was used to cluster words into 50 groups.

It is straightforward to extract such sparse features during phrase extraction. Then these features are appended to the standard phrase table for future use.

Including the sparse features drastically increases the parameter space of our systems. MERT (Och, 2003) is the default tuning algorithm for SMT. However it's not suitable for such a large scale task. Therefore, we used batch MIRA (Cherry and Foster, 2012) for tuning, which is now available in the Moses toolkit.

1.3 Operation Sequence Model

The Operation Sequence Model (OSM) explains the translation procedure as a linear sequence of operations which generates source and target sentences in parallel. Durrani et al. (2011) defined four translation operations: Generate(X,Y), Continue Source Concept, Generate Source Only (X) and Generate Identical, as well as three reordering operations: Insert Gap, Jump Back(W) and Jump Forward. These operations are described as follows.

- Generate(X,Y) make the words in Y and the first word in X added to target and source string respectively.
- Continue Source Concept adds the word in the queue from Generate(X,Y) to the source string.
- Generate Source Only (X) puts X in the source string at the current position.
- Generate Identical generates the same word for both sides.
- Insert Gap inserts a gap in the source side for future use.
- Jump Back (W) makes the position for translation be the *W*th closest gap to the current position.
- Jump Forward moves the position to the index after the right-most source word.

The probability of an operation sequence $O = (o_1 o_2 \cdots o_J)$ is:

$$p(O) = \prod_{j=1}^{J} p(o_j \mid o_{j-n+1} \cdots o_{j-1}) \qquad (2)$$

where *n* indicates the number of previous operations used.

In this paper we train a 5-order OSM and integrate this model directly into our log-linear framework with four additional features: gap penalty, open gap penalty, gap width and deletion penalty.

- Gap penalty sums to the total number of gaps inserted to produce the target sentence.
- Open gap penalty controls how quickly gaps are closed, whose value is the number of open gaps.
- Gap width calculates the distance between the first word of a source concept X and the start of the left-most gap
- Deletion penalty counts of deleted source words.

OSM is now available in Moses. However, OSM can only be applied in the phrase based translation model.

3 Experimental Set-up

This section details the experimental set-up for the submitted six systems, as well as the preprocessing steps we performed, and the construction of the language model.

1.4 Corpus and Pre-processing

All training and development corpora were provided by the organizer. The corpus was first processed using our in-house corpus processing pipeline, which removes sentence pairs that contain nonprintable characters, deletes duplicated sentence pairs, removes sentences where the number of tokens is greater than a threshold, and normalizes characters between full-width and half width characters. We also perform some language dependent preprocessing: English sentences are trucased, and if English is the target language, de-truecasing for English is also performed. Minority languages are tokenized by the organizer.

		CE news	EC news	EC s&t	TC	UC	MC
translation model	Phrase-based	✓				✓	✓
	Hierarchical Phrase-based		✓	✓	✓		
lexical reordering	Word-based	✓				✓	✓
	Hierarchical	✓				✓	
alignment	grow-diag-final-and	✓	✓	✓			
	Intersect				✓	✓	✓
max. phrase length		Default	Default	Default	Default	20	20
+multi-aligner					✓	✓	✓
+Sparse feature			✓	✓	✓	✓	
+OSM		✓				✓	
+time/number		✓	✓				

Table 3: Setting of six systems

We use the Stanford Segmenter to tokenize Chinese sentences (except for the Tibetan-to-Chinese task, where we used the segmented Chinese corpus from the baseline system provided by the organizer).

For Uyghur-to-Chinese task, we implemented a dictionary based word stemmer which is applied to the Uyghur sentences. The dictionary of the implemented stemmer is derived from the processed corpus in the baseline provided by the organizer.

Table 2 gives the statistics of corpus used in our systems.

Task	Training	Development
CE news	3,861,723	1,006
EC news	3,289,497	1,000
EC s&t	870,057	1,116
TC	109,356	650
UC	109,703	700
MC	264,922	1,000

Table 2: the number of sentences of corpus for training and tuning

The language models in our systems are trained with SRILM (Stolcke, 2002). We trained a 5-gram model with Kneser-Ney discounting (Chen and Goodman, 1996). The statistics of corpus for language model is shown in table 4. Note that there are two language models in UC task. The first model used in baseline is trained on target side of training corpus, and the second one uses Sogou12 news corpus.

Task	Sent.
CE news	9,340,957
EC news	5,463,795
EC s&t	911,526
TC	2,939,099
UC	109,703
	+1,091,494 (Sogou12)
MC	264,922

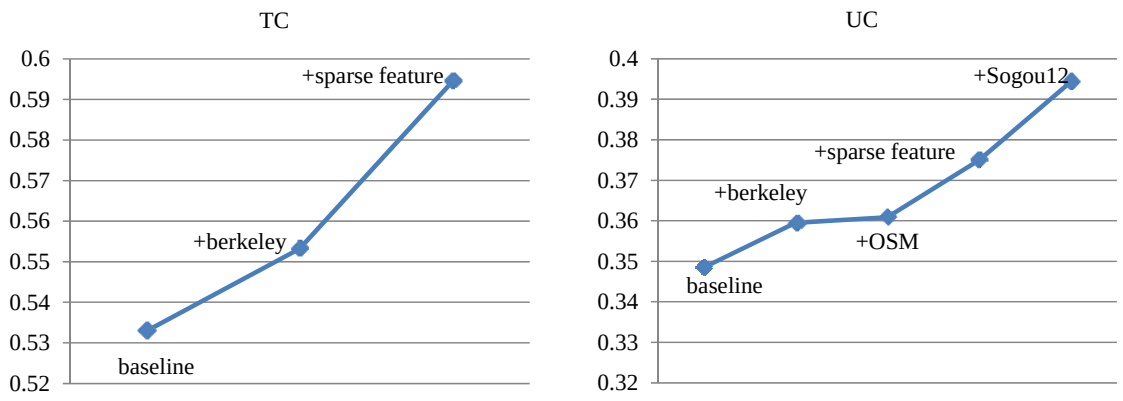
Table 4: the number of sentences of corpus for language model

1.5 Submission Systems

All the submitted systems are trained using freely available Moses toolkit (Koehn et al, 2007). Table 3 lists the features of each system. In table 3, "Default" indicates the default settings in Moses. Here we will give a brief explanation of the differences in the systems.

For the CE news and EC news tasks, we perform rule-based time/number recognition and translation only for the test set. The recognised translation pairs are directly added into phrase table with higher translation probabilities.

Where the phrase-based model is used, a lexicalized reordering model is also applied. In our systems, word-based reordering model and hierarchical reordering model (Galley and Manning, 2008) are applied with the Moses settings "*wbe-msd-bidirectional-fe*" and "*hier-mslr-bidirectional-fe*" respectively.



(a) TC (b) UC

Figure 1: effect of each techniques on TC and UC tasks

For the English-related tasks, alignment is obtained by the heuristic function "*grow-diag-final-and*". For the minority language tasks, "*intersect*" is adopted and maximum phrase length is set to 20 for phrase-based model, as they have a much smaller corpus for training. Also, these tasks use multiple phrase extraction and Good-Turing smoothing to make a more reliable translation estimation.

1.6 Experimental Results

All systems are evaluated with respect to the BLEU [Papineni et al., 2002] score. Table 5 gives the performance of individual systems on development sets. We can see from this table that the techniques described in this paper significantly improve performance on on almost all tasks.

Tasks	Dev. Set (BLEU4)	
	baseline	+techniques
CE news	0.2785	0.2858
EC news	0.2625	0.2662
EC s&t	0.4081	0.4169
TC	0.5329	0.5903
UC	0.3485	0.3944
MC	0.4408	0.4669

Table 5: comparison of performances of systems on development set before/after adopting techniques

Because we more than one technique is employed for the TC and UC tasks, Figure 1 shows how much effect each technique contributes to the systems. As more techniques are integrated, the performace improves.

4 Conclusions and Further Work

In this paper we briefly introduced our MT systems developed for and submitted to CWMT'2013. We used Moses toolkit as our machine translation system training tool and developed six baseline systems. Then we experimented with three other techniques, including multiple phrase extraction, sparse features for rule or phrase pair selection and the operation sequence model, achieving significant improvement in translation quality. In addition, rule-based time/number recognition were also performed on the CE news and EC news tasks.

However, experimental results also show inconsistency: different performance on different tasks is obtained by utilizing the same technique. Our future work will explore the causes behind this disparity, and design more powerful methods to improve systems consistently.

References

Chen, S. F. and Goodman, J. (1996). An Empirical Study of Smoothing Techniques for Language Modeling. In 34th Annual Meeting of the Association for Computational Linguistics, pp. 310–318, San Francisco, CA.

Cherry, C. and Foster, G. (2012). Batch tuning strategies for statistical machine translation. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 427-436, Stroudsburg, PA, USA.

Chiang, D. (2005). A Hierarchical Phrase-Based Model for Statistical Machine Translation. In 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), pp. 263–270, Ann Arbor, MI.

Durrani, N., Fraser, A., Schmid, H. Sajjad, H. and Farkas, L. (2013). Munich-Edinburgh-Stuttgart Submissions of OSM Systems at WMT13. In Proceedings of the Eighth Workshop on Statistical Machine Translation of ACL, pp. 122-127, Sofia, Bulgaria.

Durrani, N., Schmid, H. and Fraser, A. (2011). A joint sequence translation model with integrated reordering. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 1045-1054, Stroudsburg, PA, USA.

Galley, M., Manning, C.D. (2008). A simple and effective hierarchical phrase reordering model. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 848-856, Honolulu, Hawaii.

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

Liang, P., Taskar, B. and Klein, D. (2006). Alignment by agreement. In Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp. 104-111, New York.

Och, F. (2003). Minimum error rate training in statistical machine translation. In 41st Annual Meeting of the Association for Computational Linguistics, pp. 160–167, Sapporo, Japan.

Och, F., Ney, H. (2003). A Systematic Comparison of Various Statistical Alignment Models", Computational Linguistics, volume 29, number 1, pp. 19-51.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. In 40th Annual Meeting of the Association for Computational Linguistics (ACL-02), pp. 311–318, Philadelphia, PA.

Schütze, H. (2013). The operation sequence model: Integrating translation and reordering operations in a single left-to-right model. In Proceedings of the XIV Machine Translation Summit (Invited Talk), p. 1, Nice, France.

Stolcke, A. (2002). SRILM - An Extensible Language Modeling Toolkit. In Proceedings of the International Conference Spoken Language Processing, pp. 901–904, Denver, CO.

Zong, C. (2008) Statistical Natural Language Processing, Tsinghua Press, Beijing, China.