The *NiuTrans* Machine Translation System for NTCIR-9 PatentMT

Tong Xiao
Natural Language Processing Lab
Northeastern University
xiaotong@mail.neu.edu.cn

Qiang Li
Natural Language Processing Lab
Northeastern University
liqiangneu@gmail.com

Qi Lu
Natural Language Processing Lab
Northeastern University
luqilucy@gmail.com

Hao Zhang
Natural Language Processing Lab
Northeastern University
zhanghao1216@gmail.com

Haibo Ding
Natural Language Processing Lab
Northeastern University
dinghb@ics.neu.edu.cn

Shujie Yao
Natural Language Processing Lab
Northeastern University
yaosj@ics.neu.edu.cn

Xiaoming Xu
Natural Language Processing Lab
Northeastern University
xuxm@ics.neu.edu.cn

Xiaoxu Fei
Natural Language Processing Lab
Northeastern University
feixx@ics.neu.edu.cn

Jingbo Zhu
Natural Language Processing Lab
Northeastern University
zhujingbo@mail.neu.edu.cn

Feiliang Ren
Natural Language Processing Lab
Northeastern University
renfeiliang@ise.neu.edu.cn

Huizhen Wang
Natural Language Processing Lab
Northeastern University
wanghuizhen@mail.neu.edu.cn

ABSTRACT
This paper describes the NiuTrans system developed by the Natural Language Processing Lab at Northeastern University for the NTCIR-9 Patent Machine Translation task (NTCIR-9 PatentMT). We present our submissions to the two tracks of NTCIR-9 PatentMT, and show several improvements to our phrase-based Statistical MT engine, including: a hybrid reordering model, large-scale language modeling, and combination of Statistical approaches and Example-based approaches for patent MT. In addition, we investigate the issue of using additional large-scale out-domain data to improve patent translation systems.

Categories and Subject Descriptors
I.2.7 [**Artificial Intelligence**]: Natural Language Processing – *Machine Translation*

General Terms
Design, Experimentation

Keywords
Statistical Machine Translation, Patent Translation

TeamName: [NEU]

Subtasks/Languages: [Chinese-to-English, Japanese-to-English]

External Resources Used: [Giza++, MeCab]

1. INTRODUCTION
We describe the *NiuTrans* system submitted to the NTCIR-9 Patent Machine Translation task by the Natural Language Processing Lab at Northeastern University. Our submissions were generated using the phrase-based translation system implemented under the NiuTrans project[1]. To fit the patent translation task, our system is improved in several ways.

Some of our improvements focus on reordering in phrase-based SMT. Unlike traditional approaches, we did not resort to a single reordering model, but instead used a hybrid approach that makes use of multiple reordering models. Also, we developed a simple and fast language model for *n*-gram scoring on very large patent data, and trained a 5-gram language model using all English data (57 GB raw text) provided within the task. Moreover, we further improved our system by combining both SMT and EBMT approaches. Experimental result shows that the combined system outperformed our single SMT system over 0.4 BLEU points on the Chinese/Japanese-English patent translation evaluation data.

All our improvements resulted in seventeen features for the submitted systems. While some of the features are not new at all, we describe them (as well as corresponding feature weights) in detail. We hope that the description could ease the reproducing of our results in related tasks. Also, we present various settings of our submitted system, list the data sets used, and present both automatic and human results on the NTCIR-9 formal-run data.

In addition, we describe further experiments to study the effects of using large-scale out-domain data on patent translation. In particular, we used millions of bilingual sentences from NIST and CWMT news-domain MT tracks, as well as a very large multi-domain bilingual dictionary (over 2 million entries) to enhance the system trained using the patent data constrained to NTCIR-9 task. Interestingly, it is observed that the large-scale out-domain data is not really helpful in improving patent MT.

[1] http://www.nlplab.com/NiuPlan/NiuTrans.html

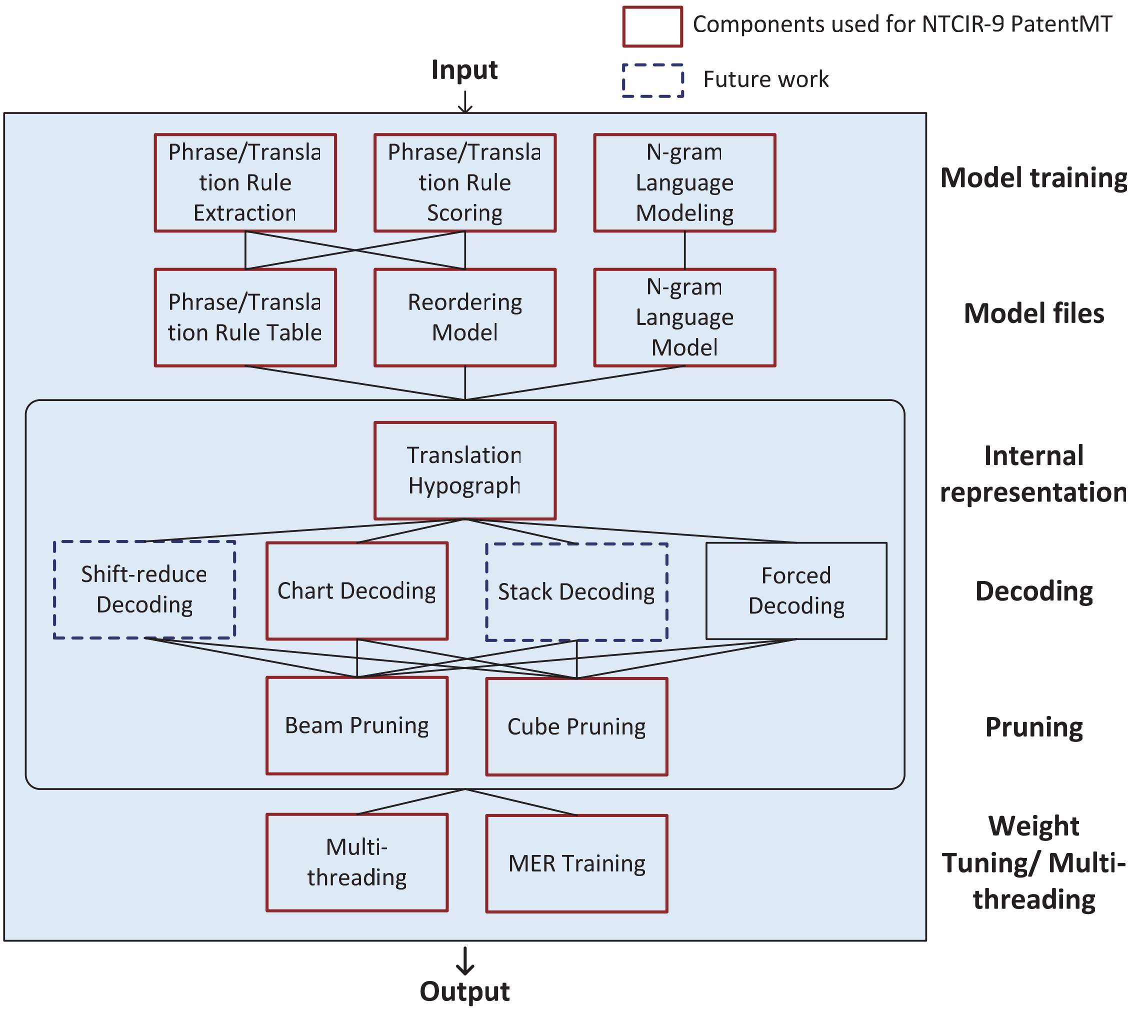

Figure 1. Architecture of the NiuTrans system.

2. THE *NiuTrans* SYSTEM

NiuTrans is an open-source Statistical Machine Translation (SMT) system. Currently it supports several state-of-the-art SMT models, including: phrase-based model, hierarchical phrase-based model and various syntax-based models (string-to-tree/tree-to-string/tree-to-tree). Figure 1 shows the architecture of the Niutrans system. In this design, all translation engines share the same internal representation of translation hypograph. Thus all the translation models can be implemented in one decoding paradigm, which provides obvious advantages for both research-oriented experiments and industrial applications. Among these engines, we chose the phrase-based engine (NiuTrans.Phrase) for our NTCIR-9 submissions due to its high speed and robustness. We did not use the syntax-based engines in this task because current parsing accuracy is far from satisfactory on non-news domain technical documents.

The NiuTrans.Phrase system follows the general framework of phrase-based MT [5] which models the translation process on non-syntactic word sequences instead of unigrams. Particularly, we focused on developing the system based on the Bracketing Transduction Grammar (BTG) [13]. Generally, three types of rules are defined in BTGs, as follows:

$$X \rightarrow \quad X_1\, X_2, \quad X_1\, X_2 \qquad \text{(R1)}$$

$$X \rightarrow \quad X_1\, X_2, \quad X_2\, X_1 \qquad \text{(R2)}$$

$$X \rightarrow \quad f, \quad e \qquad \text{(R3)}$$

where X is the only non-terminal used in BTG. R1 indicates the monotonic translation which merges two blocks (phrase pairs) into a larger block in the straight order, while R2 merges them in the inverted order. R3 is the lexical translation rule. Figure 2 shows an example where a source string is transformed into a target string using a derivation of BTG rules.

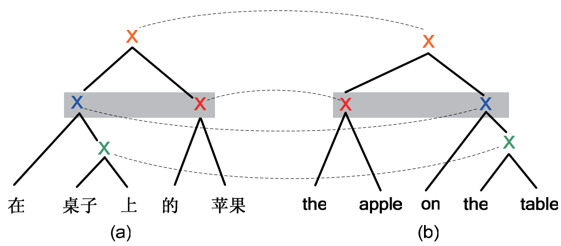

Figure 2. A sample derivation of BTG rules

Table 1. Features Used in NiuTrans.Phrase for NTCIR-9 PatentMT

	Feature	Description	Weight (*ch-en*)	Weight (*jp-en*)
1	$\Pr(t \mid s)$	Phrase trans-probability	0.089	0.107
2	$\Pr_{lex}(t \mid s)$	Lexical weight	0.043	0.034
3	$\Pr(s \mid t)$	Inverted $\Pr(t \mid s)$	0.017	0.050
4	$\Pr_{lex}(s \mid t)$	Inverted $\Pr_{lex}(t \mid s)$	0.033	0.039
5	$\Pr_{\mathrm{LM5}}(t)$	5-gram language model	0.157	0.063
6	Length(t)	# of target words	0.095	0.154
7	Count(Phr)	# of phrases	0.111	0.104
8	WD	# of word deletions	-0.006	-0.018
9	Bi-Lex	# of bi-lex links	0.082	0.051
10	Count(low-freq)	# of low-frequency rules	-0.040	-0.031
11	$f_{BTG\text{-}ME}$	ME-based reordering feature	0.193	0.201
12	$f_{M\text{-}previous}$	M orientation (previous)	0.037	0.024
13	$f_{S\text{-}previous}$	S orientation (previous)	0.017	0.014
14	$f_{D\text{-}previous}$	D orientation (previous)	0.018	0.030
15	$f_{M\text{-}following}$	M orientation (following)	0.017	0.031
16	$f_{S\text{-}following}$	S orientation (following)	0.036	0.011
17	$f_{D\text{-}following}$	D orientation (following)	0.002	0.028

Under the BTG scheme, all possible reorderings are compactly represented with binary bracketing constraints, and the decoder can be easily implemented using the chart-parsing algorithm. In NiuTrans.Phrase, all phrases are compatible with word alignments and learned using the popular method described in [5]. Currently it supports two reordering models, including: the maximum entropy-based lexicalized reordering model proposed in [14] and the MSD model proposed in [2][6][12]. We used a CKY-style decoder with cube pruning and beam search to decode new sentences under the BTG constraint. By default, the beam width was set to 30[2], and the reordering limit was set to 10. In addition to the reordering features, the NiuTrans.Phrase system adopts all standard features used in state-of-the-art SMT systems such as Moses [6]. Table 1 gives a description of the features adopted in our submitted system for NTCIR-9 PatentMT. All these features were combined in a log-linear fashion [7], and were optimized on a development data-set using the standard MERT program [8].

3. IMPROVEMENTS FOR PATENT TRANSLATION

3.1 A Hybrid Reordering Model

(Word) reordering is an old topic in machine translation. In general, we need to search for a good reordering, even if we know the correct translation for each individual word/phrase in a source sentence. This is especially important for translation tasks between languages where word orders are significant different. To date, several reordering models have been developed, showing state-of-the-art performance for many language pairs [4][6][9][12][14][15]. Although these systems and approaches are of competitive translation quality, they have different strengths and weaknesses. Take the NiuTrans.Phrase system for instance. The MSD-based reordering approach [2][6][12] is very powerful in local reordering that is inherent in the phrase translations, but has limited capabilities in dealing with the long distance dependencies. On the other hand, ME-based lexicalized reordering approach [14] characterizes the movement of hierarchical structures by phrase boundary features, but suffer from lack of local contexts in phrases.

Therefore, it is natural to explore approaches that use or combine multiple reordering approaches modeled in different views. To this end, we developed a very simple solution: all reordering models (features) were jointly used during decoding. In our case, both the ME-based lexicalized model and MSD-based model were employed together, and the corresponding features weights were jointly optimized using MERT[3]. Table 1 shows the resulting weights (h11-h17) for the submitted systems. Obviously, both the two reordering models have positive effects to our systems.

3.2 Large-scale N-gram Language Modeling

Large-scale language modeling is no-doubt a challenging issue in Machine Translation and related tasks (such as speech). Especially for machine translation, (n-gram) language models have been recognized as one of the key factors to the success of modern SMT systems. In NTCIR-9 PatentMT, we face the same problem as well. For example, in Chinese-English and Japanese-English sub-tasks, over 57 GB English patent data was provided for language modeling. We used a self-developed tool to train a 5-gram language model on such a large English corpus[4]. Our LM

[2] In our implementation, beam width refers to the number edges (derivations) that are retained in each cell. This definition is different from that used in other systems such as the Chart-decoder in Moses, where beam width is defined to be the number of all edges that are accessed for a given chart cell.

[3] In principle, Och's MERT is a line-search algorithm which optimizes each individual feature dimension in an iterative manner. Although the weight training is not conducted on a "real" joint event (i.e. feature weight vector), the correlation between different features can be somehow captured in iterations that optimizes for the same objective function (e.g. BLEU). This also suggests an interesting future direction that different reordering models are jointly learnt in both/either training (from bilingual data) and/or weight tuning (from held-out development data) stage(s).

[4] This tool is available in the NiuTrans system.

builder is basically a "sorted" trie structure [10]. The key is to develop a map that implements an array of key/value pairs, guaranteeing that the keys can be accessed in sorted order. In our implementation, each node in the trie is an entry involving 4 parts: word id, n-gram probability, backoff weight, and an offset to the array of $n - 1$ grams. These cost us 128bit to store each n-gram (as well as its probability). For each n-gram query, we first obtain its offset in the uni-gram array. Then we find its bi-gram context using binary search. This procedure is repeated until we reach the n-gram array and return the corresponding probability. As a result, each lookup is linear to the number of keys and logarithmic in the number of n-grams.

In addition to the data structure design, we also prune the model using both vocabulary filtering and n-gram filtering. To filter the vocabulary, we removed invalid characters and noisy entries (containing too many control/meaningless characters). It results in a vocabulary of 1 million entries, which is 5-times smaller than its un-filtered counterpart. Also, we set frequency thresholds for learning n-grams with higher orders. E.g. we removed the bi-/tri-grams appearing less than 3 times, and removed the 4/5-grams appearing less than 4 times. Finally we obtained a model of 6.1GB in binary format.

We then used the resulting model for inference (in MT decoding). As the translation speed of SMT systems depends heavily on the access of n-gram language model, we further speed up n-gram probability requests for decoding. The idea is not new at all: the most frequently-accessed n-grams are cached in a very fast structure (e.g. hash), which has a higher priority than trie in n-gram language model access. When the MT decoder requests an n-gram (probability), the cache is checked first. If the required n-gram hits the cache, the corresponding n-gram probability is returned by the cached copy rather than re-fetching the original data in trie. This method was very effective to our system, even achieved a speed improvement of 30% in our naive implementation.

3.3 Combining SMT and EBMT

Statistical Machine Translation (SMT) and Example-based Machine Translation (EBMT) are two state-of-the-art approaches in MT, and have been intensively investigated over the last few years. However, they are usually employed in different applications due to their individual benefits. E.g. because SMT systems are robust and require very few human labors, they are generally chosen for on-line translation and other large-scale open-domain translation tasks. On the other hand, EBMT systems are very powerful in translating "similar" sentences, and thus frequently used in domain-dependent translation, such as translating user manuals. It is an obvious next-step to make use of both SMT and EBMT for further improvement.

By offsetting weaknesses with strengths of other approaches, combination is a desirable way to achieve higher translation accuracy than does any individual approach. We tried this idea in the NTCIR-9 Chinese-English PatentMT task. In addition to the NiuTrans SMT system, we developed a simple EBMT system. Given a test sentence, it first scans the training corpus and finds the most "similar" samples using the Longest Common Subsequences (LCS) algorithm. Then it generates the translation output by only deleting *unexpected target words*[5]. We used the "one-beat-all" strategy for final translation selection: if the EBMT output is trusted enough, we selected its result as the final output; otherwise, we chose the SMT output. This raises an interesting issue: how to decide which translation is better (SMT or EBMT)? A reasonable solution might be that we straightforwardly use hypothesis selection [1][11], or other methods in SMT system combination. However, it is very time consuming to develop sophisticated combination systems, especially in the time-limited MT competitions. Thus we adopted a very simple method: we only select EBMT result as the final output when the test sentence is very similar to some training samples. In this work, we set the similarity threshold to 0.9, which resulted in about 20 final translations generated by the EBMT system on the test set.

Aside from this, our method suffers from another problem that the noisy bilingual data greatly affects the translation quality of the EBMT system. E.g., we observed that the incorrect translations in training samples often lead to the undesirable output of the EBMT system. To address this problem, we cleaned up the data by filtering out sentence pairs with very low IBM model-1 scores. Using the newly-selected data, the translation accuracy was improved by human reading in spite of fewer EBMT outputs[6]. Due to the effectiveness of this approach, we also submitted the combination result of the SMT and EBMT systems for comparison. Its BLEU result will be shown in Section 4.

3.4 Preprocessing

Before system training, all bilingual and monolingual data was pre-processed in several ways.

- Chinese and Japanese sentences were segmented using the NEUNLPLab Chinese segmentation system[7] and the MeCab system[8], respectively. For processing English sentences, a rule-based English tokenizer was employed, and the case information was removed.
- For Chinese-English MT track, all number/date/time entities were generalized to be unique symbols in order to alleviate data sparseness. These entities were then translated using an additional rule-based translation engine when we decoded test sentences. Moreover, we further improved the translation result for abbreviation, formula and mathematical symbols. We used specialized translation modules to translate these entities, and then inserted their translations into the MT system.
- All sentence pairs with unreasonable target-length/source-length ratios (< 0.2 or > 5.0) were filtered out to weaken the influence of noisy data.

[5] In our case, unexpected target words are defined to be the target words aligned to the unmatched source words in the training sentence pair.

[6] This phenomenon might be due to the careless writing/translation in applying Chinese patents.

[7] http://www.nlplab.com/

[8] http://mecab.sourceforge.net/

Table 2. Datasets used

Entry		Chinese-English C/E	Japanese-English J/E	Monolingual (English)
TRAINING	SENTENCES	1.0M	3.2M	282M
	WORDS	38M/43M	116M/110M	10882M
	VOCABULARY	300K/278K	184K/195K	1M
	ALIGNMENTS	36M	58M	N/A
DEVELOPMENT	SENTENCES	1500	2000	N/A
	WORDS	55K/60K	75K/70K	N/A
TEST	SENTENCES	2000	2000	N/A
	WORDS	55K/51K	74K/63K	N/A

Table 3. Results on NTCIR-9 PatentMT Evaluation Data

Entry	Chinese-English			Japanese-English		
	adequacy	accept	BLEU4	adequacy	accept	BLEU4
NiuTrans.Phrase	**3.51**	**0.543**	0.3229	2.37	0.416	0.2440
NiuTrans.Phrase + EBMT	N/A	N/A	**0.3273**	N/A	N/A	0.2488
Baseline 1 – Moses' hiero	3.29	0.476	0.3072	2.61	0.474	**0.2895**
Baseline 2 – Moses' phrasal	2.89	N/A	0.2932	2.42	0.447	0.2861
Baseline 3 – A rule-based system	2.27	N/A	0.1075	**3.53**	**0.674**	0.1885
Baseline 4 – Google's online translation	2.96	0.42	0.2569	2.27	0.417	0.1873

- Bi-directional word alignments were performed on the bilingual sentences with GIZA++[9]. We also refined the alignments with the "grow-diag-final-both" method for obtaining symmetric word alignment.
- To recover the case information, we used the recaser in Moses SMT toolkit[10] which is based on heuristic rules and HMM models.

4. RESULTS

The recourses we used were constrained to those provided for NTCIR-9 PatentMT. In total, 1 million Chinese-English and 3 million Japanese-English sentence pairs were used. For development data, we selected 1,500 sentences from the Chinese-English MT development data-set[11], and all the sentences from the NTCIR-8 Japanese-English evaluation data-set. The English monolingual data came from the USPTO patent documents from 1993 to 2005. Table 2 shows the statistics of the data sets used in developing our systems.

After data preparation and system update, we also tried various settings on our systems, such as enlarging beam width, smoothing translation tables and filtering reordering samples with source-side syntactic structure. To try these, we ran MERT over 30 times for the sub-tasks we participated in. Then we selected the best performing systems on the development sets and submitted them for the final evaluation in NTCIR-9 PatentMT. For both the two tracks, we chose NiuTrans.Phrase as the primary system. We also submitted the combination result of NiuTrans.Phrase and our EBMT system to the Chinese-English MT track for comparison.

The evaluation results of our submissions are listed in Table 3 where adequacy and acceptability are used as the primary evaluation metrics[12]. We see, first of all, that the NiuTrans system achieved very promising result on the Chinese-English patent translation track, even outperformed its phrase-based counterpart Moses over 1.5 BLEU points. Although our system did not show improvement over the baselines on Japanese-English track, it still suggests interesting future directions due to the so many ways to improve upon the current approach. Also, the EBMT system advanced the result as expected. It obtained a further improvement of 0.4 BLEU points over the single SMT system.

We also compared our systems with other systems submitted to the same tracks [3]. We found that, although some systems obtained very different BLEU scores, their human evaluation scores were quite competitive. This is a good sign and encourages more studies on the inconsistency between BLEU and human evaluation results. On Chinese-English patent translation, our (primary) system was ranked at 2 and 3 by adequacy and acceptability, respectively. However, there seems a gap between our systems and the top ones in Japanese-English patent translation sub-task. This we attribute to the lack of syntactic analysis of Japanese sentences. A further improvement is expected when Japanese syntax is used in our system.

5. ADDITIONAL EXPERIMENT

As described above, all the submitted systems were trained only using the patent data provided for NTCIT-9 PatentMT. A natural question that arises is whether these systems can be improved if more training data is involved. The answer to this question is

9 http://code.google.com/p/giza-pp/

10 http://www.statmt.org/moses/

11 Actually 2,000 sentences were provided for system tuning in the Chinese-English translation track. Aside from the development set of 1,500 sentences, the remaining 500 sentences were chosen for our held-out test during system development.

12 Both the two scores were assigned by human judges.

Table 4. Additional Open-domain Datasets

Entry	NIST news C/E	CWMT news C/E	Multi-domain dictionary
SENTENCES/ENTRIES	2.0M	3.1M	2.0M
WORDS	49M/55M	60M/65M	N/A
VOCABULARY	209K/135K	393K/374K	N/A
ALIGNMENTS	46M	55M	N/A

Table 5. Results of Using Additional Training Data

Entry	BLEU4	
	Dev	Test
Baseline (NTCIR-9 CE PatentMT)	**0.3311**	**0.3217**
+ NIST CE news	0.3257	0.3171
+ CWMT CE news	0.3279	0.3148
+ multi-domain bi-dict	0.3282	0.3172
+ all	0.3270	0.3165

especially important when the systems are applied to real-word applications and developed in an "unconstrained" setting. Therefore, we designed an additional experiment to investigate this issue on Chinese-English patent translation. However, it is rare to see large-scale bilingual patent text which is publicly available for MT system development. Thus we used some "out-domain" data-sets instead (Table 4).

- **NIST Chinese-English MT**[13]: We selected most data-sets from those available for NIST 2008 Chinese-English "constrained training" track, except the UN Chinese-English parallel text. Most of these texts come from news report.
- **CWMT Chinese-English news translation**[14]: We also used all the data available for the CWMT 2011 Chinese-English news translation track. This dataset is a mixture of news reports, technical documents and various texts from conversation and web.
- **A large-scale multi-domain dictionary**: In addition to bilingual sentences, we also examine the effects of using large-scale bilingual dictionary on this task. To do this, we used a multi-domain Chinese-English dictionary obtained from web. It contains about 2 million entries in 60 domains.

We selected 500 sentences from the development data for held-out test. Table 5 shows the BLEU score of our SMT system (NiuTrans.Phrase) when more training data was involved. As seen from the Table, using additional out-domain data-set was not very helpful in improving our patent translation system. Even though the training data-set was scaled-up 4 times, no (significant) improvement was observed.

While our experiments did not show promising result, it still raises many interesting questions for future study, such as: how would patent MT systems benefit from non-patent training data; and how would the domain adaptation method work for patent MT when multi-domain data is used. They are worth an in-depth study in our future work.

6. CONCLUSION

We have presented our submissions to the Chinese-English and Japanese-English tracks of NTCIR-9 PatentMT. By enhancing a phrase-based SMT system (NiuTrans.Phrase), our improved system significantly outperformed the four baseline systems on the Chinese-English MT track. Also, we observed that using out-domain data was not very beneficial to the patent translation system. Although some of our results are still preliminary, a number of issues were raised and suggested interesting future directions, such as: domain adaptation for patent MT.

7. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation of China (60873091; 61073140), Specialized Research Fund for the Doctoral Program of Higher Education (20100042110031), the Fundamental Research Funds for the Central Universities, and the Natural Science Foundation of Liaoning Province of China.

8. REFERENCES

[1] Hildebrand, A. and Vogel, S. 2008. Combination of machine translation systems via hypothesis selection from combined *n*-best lists. In *Proc. of* Conference of the Association for Machine Translation in the Americas (AMTA), 254-261.

[2] Galley, M. and Manning, C. 2008. A Simple and Effective Hierarchical Phrase Reordering Model. In *Proc. of* Empirical Methods on Natural Language Processing (EMNLP) 2008, Hawaii, 848-856.

[3] Goto, I., Lu, B., Chow, K., Sumita, E. and Tsou B.. 2011. Overview of the Patent Machine Translation Task at the NTCIR-9 Workshop, In *Proc of* the 9th NTCIR Workshop.

[4] Kumar, S. and Byrne, W. 2005. Local phrase reordering models for statistical machine translation. In *Proc. of* Human Language Technology Conference and Conference on

[13] http://www.itl.nist.gov/iad/mig//tests/mt/

[14] http://nlp.ict.ac.cn/new/CWMT/index.php

Empirical Methods in Natural Language Processing (HLT/EMNLP) 2005, Vancouver, 161–168.

[5] Koehn, P., Och, F. and Marcu, D. 2003. Statistical Phrase-Based Translation. In *Proc. of* Human Language Technology and Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL) 2003, Edmonton, USA, 48-54.

[6] Koehn, P., Hoang, H., Mayne, A., 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 Annual meeting of the Association for Computational Linguistics (ACL) 2007 demonstration session.

[7] Och, F. and Ney, H. 2002. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *Proc. of* Annual meeting of the Association for Computational Linguistics (ACL) 2002, Philadelphia, 295-302.

[8] Och, F. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of* Annual meeting of the Association for Computational Linguistics (ACL) 2003, Japan, 160-167.

[9] Ohashi, K., Yamamoto, K., Saito, K., and Nagata, M. (2005). NUT-NTT statistical machine translation system for IWSLT 2005. In *Proc. of* the International Workshop on Spoken Language Translation.

[10] Pauls, A. and Klein, D. 2011. Faster and Smaller N-Gram Language Models. In *Proc. of* Annual meeting of the Association for Computational Linguistics (ACL) 2011, 258–267.

[11] Sim, K., Byrne, W., Gales, M., Sahbi, H. and Woodland, P. 2007. Consensus network decoding for statistical machine translation system combination. In *Proc. of* ICASSP 2007, 105-108.

[12] Tillmann, C. 2004. A unigram orientation model for statistical machine translation. In *Proc. of* Human Language Technology and Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL) 2004: Short Papers, 101-104.

[13] Wu, D. 1996. A polynomial-time algorithm for statistical machine translation. In *Proc. of* Annual meeting of the Association for Computational Linguistics (ACL) 1996, California, USA, 152–158.

[14] Xiong, D., Liu, Q. and Lin S.. 2006. Maximum Entropy Based Phrase Reordering Model for Statistical Machine Translation. In *Proc. of* Annual meeting of the Association for Computational Linguistics (ACL) 2006, Sydney, 521-528.

[15] Zens, R. and Ney, H. 2006. Discriminative reordering models for statistical machine translation. In *Proc. of* the Workshop on Statistical Machine Translation, New York City, 55–63.