NAME-AWARE LANGUAGE MODEL ADAPTATION AND SPARSE FEATURES FOR STATISTICAL MACHINE TRANSLATION

Wen Wang[1] *Haibo Li*[2] *Heng Ji*[3]

[1]SRI International
[2]Nuance
[3]Rensselaer Polytechnic Institute
wwang@speech.sri.com, lihaibo.c@gmail.com, jih@rpi.edu

ABSTRACT

We propose approaches improving statistical machine translation (SMT) performance, by developing name-aware language model adaptations and sparse features, in addition to extracting name-aware translation grammar and rules, adding name phrase table, and name translation driven decoding. Chinese-English translation experiments showed that our proposed approaches produce an absolute gain of +2.3 BLEU on top of our previous high-performing, name-aware machine translation system.

Index Terms— statistical machine translation, name translation, sparse features, language model adaptation

1. INTRODUCTION

It has become increasingly important to extract reliable information from the vast and multilingual streams of raw data flowing around the world. Cross-lingual information distillation, such as entity linking, event extraction, slot filling and question answer, can address part of this need. However, a key bottleneck of high-quality cross-lingual information distillation lies in the performance of statistical machine translation (SMT). For example, in cross-lingual slot filling, 59% of omission errors and 85% of spurious errors were due to MT errors on names [1]. Traditional SMT approaches focus on the fluency and accuracy of the overall translation but fall short in their ability to translate certain information units (content words that contain critical information), especially names, concepts, events and topics. Names often contribute significantly to the meaning of sentences, yet a typical state-of-the-art Chinese-English statistical MT system can only translate 60% of person names correctly [1]. Furthermore, incorrect name translations could also cause incorrect translations of long contexts.

Prior research effort on incorporating name translations into SMT systems all focused on loose coupling of the two systems. They can be largely categorized into the following two types of approaches, namely, preprocessing and postprocessing. Preprocessing approaches identify names in the source input and propose name translations to the MT systems. Then name translation results can be transferred from the source side to the target side using word alignment, or can be added to the phrase table and let the language models (LMs) decide which translations to choose [1]. [2] developed heuristic rules to create the "do-not-translate" list and [3] learned supervised models to decide when to transliterate. In contrast, post-processing approaches explore online query names in a cross-lingual information retrieval or question answering framework to obtain translations and post-edit the MT output [4] [5] [6] [7].

In our prior work [8], we developed a name-aware machine translation (NAMT) approach that is different from these prior efforts. Our NAMT approach tightly integrates name processing into the MT model, by jointly annotating parallel corpora, extracting name-aware translation grammar and rules, adding name phrase table, and exploiting name translations during decoding. The NAMT system yields consistent gains on BLEU [9] and Translation Edit Rate (TER) [10] on all test sets and up to 23.6% relative error reduction on name translation. There has been effort on employing and studying the NAMT approach for other language pairs and specific domains. For example, [11] replicated the NAMT approach using Moses for English-Spanish translations, but observed that the room for improvement on name translation accuracy is much smaller than that for Chinese-English. Based on their named entity translation error analysis, they left named entities with more than one occurrence to the SMT system to handle. For named entities with zero or one occurrence, they used a specialized module to generate translations and add them dynamically to the phrase table. This module merges results from several name translation techniques, for example, different dictionaries. They observed a small but statistically significant BLEU gain (+0.2) but no gain on name translation accuracy. On the other hand, [12] proposed ideas for porting the NAMT approach to handle names in the information technology domain.

In this work, we advance our prior research [8] by exploring named entity information in language model adaptation for MT, both for search and for N-best reranking with bilingual recurrent neural network language models, and by exploring sparse features and name-aware sparse features. The rest of the paper is organized as follows. Section 2 describes the baseline MT system. Our prior work of the name-aware machine translation approach is reviewed in Section 3. Section 4 and Section 5 present our innovations on exploring named entity information for SMT LM adaptation and sparse features, respectively. Experimental results and discussions appear in Section 6 and we conclude in Section 7.

2. BASELINE MT SYSTEM

Our baseline Chinese-English MT system, denoted *baseline-MT*, is based on the hierarchical phrase-based translation framework (Hiero) [13] using weighted synchronous context-free grammar (SCFG). All SCFG rules are associated with a set of features that are used to compute the derivation probabilities in a log-linear model [14]. We incorporate six dense features for each SCFG rule, including the IBM Model-1 lexical scores in both source-to-target and target-to-source directions, relative frequencies for bilingual SCFG rules in both directions learnt from the parallel data, phrase penalty

978-1-4799-7291-3/15/$31.00 ©2015 IEEE

for a rule with no non-terminal being used in derivation, rule penalty for a rule with at least one non-terminal being used in derivation, glue rule penalty if a glue rule is used in derivation, and translation length as the number of words in the translation output.

The baseline MT system uses a log-linear combination of multiple language models (LMs) trained on various sources using modified Kneser-Ney smoothing algorithm [15] and converted to Bloom filter LMs [16] supporting memory map. The feature weights are optimized by minimum error rate training (MERT) to maximize BLEU scores [17]. The decoding is chart-based, and the standard CKY algorithm is applied to derive translations from a packed forest.

3. BASELINE NAME-AWARE MACHINE TRANSLATION

The name-aware machine translation system we developed previously is described in [8], and this is the baseline name-aware MT system for this paper, denoted *baseline-NAMT*. We use the name tagged parallel corpora for training and name tagged test set with name translations for evaluation. For training, we run our joint name tagger on the training parallel text, replace tagged name pairs with their entity types, and then use GIZA++ and symmetrization heuristics to regenerate word alignment. We found the name tags appear very frequently and the existence of such name tags yields improvement in the word alignment quality. We merge the name-replaced parallel data with the original parallel data, and extract grammars from the combined corpus. Note that the joint name tagger ensures that each tagged source name has a corresponding translation on the target side (and vice versa), we can extract SCFG rules by treating the tagged names as non-terminals.

For decoding, names in the test sets with their frequencies fewer than five instances in the training data are translated through the name translation system; and the rest are translated by the MT system. Our decoder rewrites the non-terminals in SCFG rules into the extracted names, hence allowing unseen names in the test sets to be translated. Also, our decoder exploits the dynamically created phrase table from name translations and competes with originally extracted rules, through the LMs, to find the best translation hypothesis for the source language input. More details of this baseline-NAMT system can be found in [8].

4. NAME-AWARE LANGUAGE MODEL ADAPTATION

We develop name-aware, topic-based language model adaptation both for decoding and for N-best reranking. For decoding, we combine a dynamically adapted word n-gram LM with the baseline mixture of LMs for search. For N-best reranking, we investigate the efficacy of name-aware bilingual recurrent neural network (BiRNN) LM adaptation for N-best reranking. The general framework of conventional topic-based LM adaptation approaches is as follows. Topic analysis is conducted on documents in the LM training text collection and each document is assigned a topic among K topics. During testing, for each test set document, we identify its topic T^* through topic analysis and adapt the decoding LM or N-best reranking LM. Since we use different strategies for adapting the decoding and reranking LMs, we discuss them separately in the following subsections.

4.1. Name-aware Adaptation of Decoding Word N-gram LMs

Inspired by the work in [18], we investigate clustering-based topic analysis and LDA topic analysis for adapting the decoding word n-gram LM.

Clustering-based Topic Analysis. We use the CLUTO toolkit [1] to cluster the documents in the MT parallel training data into K clusters. CLUTO first converts the set of documents into the vector space format, i.e., representing each document as a feature vector, then finds a predefined number K of clusters based on a specific criterion. We choose the cosine-distance based metric to measure the similarity between two documents. After clustering, we train one word n-gram LM for each cluster using the documents in this cluster, resulting K topic LMs. For this work, we modified the doc2mat tool from CLUTO to correctly process Chinese documents. During decoding, for a test set document, we compute the perplexity of each topic LM $p_{t_i}(\cdot), 1 \leq i \leq K$ on this test set document where t_i is the i^{th} topic, and select the topic cluster T^* with the lowest perplexity. The corresponding $p_{T^*}(\cdot)$ is selected as the adapted LM LM_a, that is, $p_{LM_a}(w|h) = p_{T^*}(w|h)$. We then combine LM_a in log-linear interpolation with the baseline mixture of LMs, denoted LM_{bg}, for decoding, and tune the interpolation weight between LM_a and LM_{bg}, on one development set.

In the conventional clustering-based topic analysis, all words (with optional stopwords removal and stemming) are considered to construct the feature vectors for the documents. In our name-aware approach, we use tagged named entities to construct the feature vectors during training.

LDA Topic Analysis. LDA model [19] is a Bayesian extension of a mixture of unigram models where a vector of topic mixture weights θ is drawn from a prior Dirichlet distribution:

$$f(\theta;\alpha) = \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \tag{1}$$

where $\alpha = \alpha_1, \cdots, \alpha_K$ represents the prior observation count of the K latent topics and $\alpha_k > 0$. For a document $d = w_1, w_2, \cdots, w_n$, the LDA model assigns it the following probability:

$$p(w_1^n) = \int_\theta (\prod_{i=1}^{n} \sum_{k=1}^{K} \beta_{w_i,k} \cdot \theta_k) f(\theta;\alpha) d\theta \tag{2}$$

During training, the LDA analysis generates the topic mixture weights θ for each document and we choose the topic with the highest mixture weight as the topic for the certain document. Then we train a topic-specific LM using documents belonging to each topic. During decoding, for each test set document $d = w_1, w_2, \cdots, w_n$, the LDA model generates a mixture of topics and we compute the dynamically adapted LM_a as follows:

$$p_{LM_a}(w|h) = \sum_{i=1}^{K} \gamma_i \cdot p_{t_i}(w|h) \tag{3}$$

where p_{t_i} represents the i^{th} topic specific LM and γ_i is the mixture weight. We compute γ_i as follows [18]:

$$\begin{aligned} \gamma_i &= \sum_{j=1}^{n} p(t_i|w_j) p(w_j|d) \\ p(t_i|w_j) &= \frac{f_{ji}}{\sum_{p=1}^{K} f_{jp}} \\ p(w_j|d) &= \frac{freq(w_j|d)}{\sum_{q=1}^{n} freq(w_q|d)} \end{aligned} \tag{4}$$

[1] http://glaros.dtc.umn.edu/gkhome/views/cluto

where $freq(w_j|d)$ denotes the frequency of a word w_j in the document d and f_{ji} represents the frequency of a word w_j generated from a topic $t_i (i = 1, 2, \cdots, K)$ over the training data, based on the LDA analysis.

In this work, we employ the Java implementation of LDA using Gibbs sampling for parameter estimation and inference [2], for LDA topic analysis. Note that in our name-aware approach, we use tagged named entities to represent the documents, instead of considering all words as in the conventional LDA topic analysis.

4.2. Name-aware Adaptation of Bilingual RNN LMs

A recurrent neural network language model (RNNLM) [20] captures complex long-distance history across sentence boundaries. Given a sentence $E = e_1 \cdots e_j \cdots e_J$, it predicts the current word e_j given the previous word e_{j-1} and the previous hidden state vector h_{j-1}. A bilingual recurrent neural network (BiRNN) LM is proposed in [21] and it produced significant gain on MT performance compared to using monolingual RNN LMs (i.e., standard RNNLMs trained on the monolingual data in the target language) for reranking N-best lists. BiRNNLMs are trained on parallel corpora and have stronger word prediction power than RNNLMs trained on monolingual corpora, since for BiRNNLMs, word prediction can be conditioned on the complete source sentence in addition to the previous target word and hidden state vector, whereas monolingual RNNLMs only condition word predictions based on the previous word and hidden state vector. In other words, given a set of parallel sentences and corresponding word alignment F, E, A where $F = f_1 \cdots f_i \cdots f_I$, $E = e_1 \cdots e_j \cdots e_J$, and $A = a_1 \cdots a_j \cdots a_J$ denote a source sentence, target sentence, and a word alignment respectively, the standard monolingual RNN LM for the target side is $p(e_j|e_{j-1}, h_{j-1})$; whereas, the BiRNN LM can be represented as $p(e_j|e_{j-1}, h_{j-1}, F, A)$.

For developing our BiRNN LMs, a "bag-of-word" (BOW) approach [21] is used on a source sentence to construct a sparse input feature vector. The BOW approach does not consider word frequencies in order to minimize the effect of frequent function words like "the". To minimize direct modeling of source-to-target word translations, which has already been covered by IBM-1 model, i.e. $p(e|f)$, the "less-one" approach is employed, in which we deactivate the source words that are direct translations to the current and previous target words according to the word alignment. These translation pairs may have been well captured by IBM translation models that are used as features in the log-linear framework. Also, these aligned source words may significantly reduce the impact of other predictors, such as the previous target word predictor. In this work, the BiRNN LMs are all trained with the BOW and less-one approaches.

We use the aligned sentence pairs following their orders in the original training documents, for training BiRNN LMs. We selected 10% of the training data for cross-validation on word perplexity to set the learning rates and to decide when to terminate the training. We use 100 output classes for the class layer in BiRNN to reduce computational complexity, similar to factorizing the output layer with the class layer in the RNNLM [22]. Words in the target language vocabulary are assigned to the 100 classes based on their frequencies, i.e., through frequency binning. We use the backpropagation through time algorithm [20] for training the weights. Initial learning rate is chosen as 0.1, with the learning rate started to halve when the perplexity reduction on the validation set in cross-validation is small.

The adaptation process for the BiRNN model is a one-iteration retraining using the source input and their 1-best translation output from the rescored n-best lists. We extend this approach to a topic-based BiRNNLM training and adaptation paradigm. For BiRNN LM, during training, we train K BiRNNLMs based on parallel documents belonging to each topic. The topic assignment for each parallel document is from the clustering-based topic analysis or the LDA analysis on the source side, described above. During testing, we assign each test set document its topic based on topic analysis on the source input and select the corresponding topic-specific BiRNNLM for rescoring. Then we adapt the topic-specific BiRNNLM based on the parallel text of the source input and their 1-best translation output from rescored N-best lists from test documents belonging to this certain topic, instead of adapting based on 1-best output from rescored N-best lists on the entire test set. The learning rate of BiRNNLM adaptation is determined based on optimizing the perplexity on the development test set.

For name-aware adaptation of BiRNNLMs for reranking, similar to the approaches for adapting word n-gram LMs for decoding, the topic analysis is conducted on tagged named entities instead of all words for documents.

5. SPARSE FEATURES

Statistical machine translation decoding follows a log-linear framework for translating a foreign language F into English E, as shown in Equation 5.

$$E^* = \arg\max_{E} P(E|F) \approx \arg\max \vec{\lambda} \cdot \vec{h}(F, E) \qquad (5)$$

where $\vec{h}(F, E)$ is a vector of features defined for a given source sentence and target sentence pair: F, E. The feature weights, denoted $\vec{\lambda}$, are learned from a tuning set of sentence pairs. Traditionally SMT systems only use the order of tens of dense features. Recent introductions of large-scale tuning algorithms such as MIRA [23] or tuning-as-reranking (PRO) [24] enable MT systems to use a large amount of features. Sparse features are designed to fix specific machine translation errors. For example, a lexical error can be fixed by checking a particular word-pair and assigning a weighted penalty to the error. We add seven categories of sparse features to help check specific evidences in the SCFG rules and fix errors in the SMT output [25], as shown in Table 1. Among these categories, the rule type category is a more detailed description of SCFG rules. We use 38 rules types in this work. In this work, on our data set, we start with a list of 1,489,960 sparse features based on the seven categories. Most of the sparse features are simple binary-valued features. Many sparse features are very specific. Also, the majority of the features overlap with each other, for example, the Bigram features overlap with word n-gram language model scores. Optimizing this large amount of over-lapping features could easily get over-fitted to the tuning set.

We employ the tuning-as-reranking algorithm [24] for optimizing the sparse features. We use the linear support vector machine (SVM) classifier using a L2 regularizer to learn the weights. To alleviate the overfitting problem for optimizing sparse features, the weights learned for each feature are used for feature selection, that is, all the features with a weight close to zero will be removed from the feature list, and the remaining features are used for the next optimization iteration and so on. In these iterations, we only kept the top 5K weighted sparse features. For learning the weights for both dense features and sparse features, we use a two-stage optimization strategy. We first optimize the weights for the dense features. Then we optimize the weights for the 5K sparse features together with the

[2] http://jgibblda.sourceforge.net/

dense features while fixing the weights learned earlier for the dense features. In our experiments, we find this approach of feature selection and two-stage optimization is most stable and produces the best BLEU and TER gain over the dense features. Table 2 shows examples of selected spare features and their weights.

We then add a set of name-aware sparse features on top of the 5K selected sparse features, including

- name(f, NULL) fires when a source input name f is dropped
- name(NULL, e) fires when a target name e is generated unaligned
- name(f, $\bar{ne}$) fires when a source input name f translates to a non-name word $\bar{ne}$
- name($\bar{ne}$, e) fires when a source non-name word $\bar{ne}$ translates to a target name e
- name(match(f,e)) fires when a source input name f translates to a target name e with matched name types
- name(mismatch(f,e)) fires when a source input name f translates to a target name e with mismatched name types
- name types in rules, i.e., enriched Hiero rule types with name types
- Binned frequency of enriched Hiero rule types with name types

Table 2. Example selected sparse features and their weights.

Features	Weights
F-X-F-X-F → X-E-X	0.1210
X-F-X → E-X-E-X-E	0.2686
X1X0W	-0.10161
BI_most_awesome	0.15436

6. EXPERIMENTS

We present the experimental results from name-aware language model adaptation and sparse features, compared to the baseline MT and the baseline-NAMT systems.

6.1. Data and Experimental Setup

We use exactly the same Chinese-English MT training data and the tune set as in [8] for training the MT system and optimizing parameters. For evaluation, we use the NIST part of the 2006 and 2008 NIST openMT evaluation test sets, denoted *NIST2006* and *NIST2008* test sets. The large training data covers various sources and genres, including newswire, web text, broadcast news and conversation transcripts. We also used some translation lexicon and Wikipedia translations. The majority of the parallel training data were released by LDC for U.S. DARPA Translingual Information Detection, Extraction and Summarization (TIDES) program, Global Autonomous Language Exploitation (GALE) program, and Broad Operational Language Translation (BOLT) program, and National Institute of Standards and Technology (NIST) open MT evaluations. The name tagged parallel training data includes 1,686,458 sentence pairs. 1,890,335 name pairs were tagged (295,087 Persons, 1,269,056 Geopolitical entities, and 326,192 Organizations). The decoding LM is a log-linear combination of four word n-gram LMs, trained from different English corpora, including the target side of BOLT parallel text, English monolingual discussion forums data R1-4 released in BOLT Phase 1 (LDC2012E04, LDC2012E16, LDC2012E21, and LDC2012E54), English Gigaword Fifth Edition (LDC2011T07), the web text portion of the parallel text, and the broadcast news and conversation transcripts released under the DARPA GALE program.

For generating training documents for topic analysis described in Section 4, we use document boundaries for a corpus when the information is released with the corpus. For a corpus without document boundaries, we simply split them into shorter segments and treat each segment as a "document". In total, we have 48,975 documents based on the MT training data. We use these as the documents for clustering-based and LDA topic analysis during training. For this work, we set $K = 50$ as the predefined number of topics for topic analysis, which is empirically chosen based on the perplexity on the development set from the adapted LM, with a few K values from 10 to 100.

MT feature weights, including sparse features, are tuned on a set of 2,770 sentences. The efficacy of sparse features and name-aware sparse-features is evaluated on the NIST2006 and NIST2008 test sets. For investigating the efficacy of name-aware language model adaptation, the log-linear interpolation weight λ between the baseline LM LM_{bg} and the adapted LM LM_a is tuned on the NIST2006 test set by a grid search. Then the resulting λ is used for evaluating the LM adaptation performance on the NIST2008 test set. The LDC releases of the NIST part of the NIST2006 (LDC2010T17) and NIST2008 (LDC2010T21) test sets include 79 and 109 documents, respectively. We manually adjusted the document boundaries for the NIST2006 test set into 83 documents.

For the tune set, NIST2006, and NIST2008 test sets, we extract names with the baseline monolingual name tagger described in [26] from the source documents. The performance of this monolingual name tagger is comparable to the best reported results on Chinese name tagging on Automatic Content Extraction (ACE) data [26]. Then we apply a state-of-the-art name translation system [1] to translate names into the target language. As in [8], this name translation system was enhanced by a name origin classifier based on Chinese last name list (446 name characters) and name structure parsing features to distinguish Chinese person names and foreign person names, so that pinyin is applied for Chinese names whereas name transliteration is applied for foreign names. For the extracted names from test sets, a joint bilingual name tagger [26] was employed to mine bilingual name translation pairs from the parallel training data. The automatically mined unique name translation pairs were used to create a name phrase table, which was added for MT decoding.

Table 3 summarizes the statistics of the name tagged NIST2006 and NIST2008 test sets with name translations. Both NIST2006 and NIST2008 test sets are composed of documents from newswire, broadcast news, and web blogs.

6.2. Results

Table 4 shows the BLEU scores from adapting the decoding word n-gram LMs using clustering-based (denoted *CL*) and LDA (denoted *LDA*) topic analysis, comparing between considering all words (denoted CL,LDA(All)) and considering only named entities (denoted CL,LDA(NE)) for topic analysis. The BLEU scores from the baseline MT and baseline-NAMT systems are also shown.

As can be seen from Table 4, combining the baseline LMs with an adapted LM always improves BLEU scores. LM adaptation based on the clustering-based topic analysis yields better BLEU scores

Table 1. Sparse Feature Types and Descriptions.

Feature Categories	**Description**
Lexical	If the word-pair $f \leftrightarrow e$ occurs in the derivation from the IBM model-1.
Fertility	The number of times a word is aligned to 1 word, 2 words, or 3+ words.
Rule type	Detailed Hiero rule types (e.g., F-X1-F-X2 $\leftrightarrow$ X1-E-X2).
Reorder type	If the target side contains monotone or reordering of non-terminals (e.g., X1X0W).
Target spontaneous words	Pre-defined English spontaneous words (e.g., this, the, such)
Bigrams	Bigrams seen in the target side of the phrases (e.g., $BI\_w_1\_w_2$)
Frequency of rules	Binned frequency of the observed rules

Table 3. Statistics and name percentages of the NIST2006 and NIST2008 test sets.

Test set	**#Documents**	**#Sentences**	**#Words in Source**	**#Words in Refs**	**#All names (%occurred < 5)**
NIST2006	83	1,664	38,442	45,914	2,853 (73.1)
NIST2008	109	1,357	32,646	37,315	1,462 (72.0)

than using LDA topic analysis. And replacing all words with only named entities for topic analysis produces better BLEU scores for both CL and LDA topic analysis approaches. It is worth noting that the baseline LMs are trained on much larger data than the MT training parallel data, whereas the adapted LM is either a topic-specific LM (the CL approaches) or a mixture of topic-specific LMs (the LDA approaches) based on documents from the MT training parallel data. So it is encouraging to see the gain from combining this much smaller adapted LM_a with the baseline LMs. LM_a from CL(NE) produces the best BLEU gain, +0.5 and +0.4 absolute, for the NIST2006 and NIST2008 test sets. We use this configuration to dump 2000-best lists for investigating the efficacy of name-aware adaptation of BiRNNLM N-best reranking.

Table 5 shows the BLEU scores from applying bilingual RNNLM (BiRNNLM) for reranking 2000-best lists from the best configuration in Table 4. In this work, we set the number of hidden unit as 600 for training the BiRNN LMs. Again, we compare between using a single BiRNNLM trained on the entire MT training data, and training K BiRNNLMs for the K topic clusters from CL and LDA topic analysis approaches and adapting a BiRNNLM for the i^{th} topic on test set documents belonging to i^{th} topic, as described in Section 4. We also compare between considering all words and only named entities for topic analysis. We observe applying a single BiRNNLM for reranking produces +0.4 and +0.3 absolute BLEU gain on NIST2006 and NIST2008 test sets. However, when adapting this BiRNNLM on the entire test set and reranking the test set again, i.e., the row of (2) + single BiRNNLM-adapt, it doesn't change BLEU on the NIST2006 test set and hurts slightly on NIST2008 test set. The following four rows correspond to the topic-specific BiRNNLM adaptation approach described in Section 4. We observe that the LDA(All) adaptation approach slightly hurts BiRNNLM reranking performance, whereas CL(All) produces a small BLEU gain. The LDA(NE) adaptation approach produces only +0.1 BLEU gains from adapting BiRNNLMs, but CL(NE) produces +0.4 BLEU gain, compared to using the single, unadapted BiRNNLM for reranking. For both adapting decoding word n-gram LM and reranking BiRNNLM, we observe that using named entity information can produce more consistent improvement on the LM adaptation performance than considering all words for topic analysis, for both clustering and LDA approaches. Since topic analysis could be noisy, we also combine the reranking scores from the single BiRNNLM trained on the entire MT training data with the CL(NE) adapted topic-specific BiRNNLM reranking scores, and this yields another +0.2 BLEU gain on both NIST2006 and NIST2008 test sets. Combining name-aware LM adaptation for decoding LM and bilingual RNNLM for N-best reranking, we achieve an improvement of **+1.5** and **+1.3** BLEU on the NIST2006 and NIST2008 test sets (i.e., 37.8 and 31.3 BLEU over 36.3 and 30.0 BLEU, for NIST2006 and NIST2008 respectively).

Table 6 shows the efficacy from sparse features, adding name-aware sparse features, and adding the best configuration of name-aware LM adaptation (shown in Table 5). The selected 5K sparse features out of the seven types produce +0.7 and +0.8 better BLEUs on the NIST2006 and NIST2008 test sets. Adding name-aware sparse features yields +0.4 and +0.3 BLEU gain on the two test sets. Note that the weights of the sparse features are optimized on the 2,770-sentence tune set. Finally, employing the best configuration of name-aware LM adaptation for both decoding LM and BiRNNLM reranking (the configuration as the last row in Table 5) yields an additional +1.0 and +1.2 BLEU improvement on the two test sets. Overall, the name-aware sparse features and name-aware LM adaptation proposed in this work achieve **+1.4** and **+1.5** BLEU gain on the NIST2006 and NIST2008 test sets, over the second row of baseline-NAMT+sparse features, and achieve **+2.1** and **+2.3** BLEU gain over our name-aware MT performance published in [8]. These four BLEU gains are statistically significant at $p = 0.05$, using a paired bootstrap resampling test [27].

7. CONCLUSION

We investigated incorporating named entity information into language model adaptation and sparse features for improving statistical machine translation. Chinese-English SMT experiments have shown that using named entities for topic analysis is always more stable for the performance of adapted LMs, compared to considering all words. And we obtained consistent gains on BLEU from name-aware adaptation of the decoding word n-gram LMs and the bilingual RNN LMs for N-best reranking, using the clustering-based topic analysis approach. Furthermore, the gains from sparse features and name-aware sparse features, and from name-aware LM adaptation are additive, yielding an overall +2.3 absolute BLEU gain on the test set.

Table 4. BLEU scores on the NIST2006 and NIST2008 test sets exploring name-aware language model adaptation for decoding word n-gram LM, compared to the baseline MT system and the baseline-NAMT system. CL denotes the clustering-based topic analysis and LDA denotes LDA topic analysis. CL,LDA(All) denote topic analysis considering all words as in the conventional approaches, whereas CL,LDA(NE) denote topic analysis only considering named entities.

		BLEU	
		NIST2006	NIST2008
(1)	baseline MT	35.5	29.3
(2) Adapt decoding LM	baseline-NAMT (Li et al., ACL 2013)	36.3	30.0
	(2) + CL(All)	36.5	30.3
	(2) + LDA(All)	36.4	30.2
	(2) + CL(NE)	**36.8**	**30.4**
	(2) + LDA(NE)	36.6	30.4

Table 5. BLEU scores on the NIST2006 and NIST2008 test sets exploring name-aware language model adaptation for bilingual RNNLM (BiRNNLM) N-best reranking. CL denotes the clustering-based topic analysis and LDA denotes LDA topic analysis. CL,LDA(All) denote topic analysis considering all words in a document as in the conventional approaches, whereas CL,LDA(NE) denote topic analysis only considering named entities. Topic analysis was used for adapting the K topic-specific BiRNNLMs.

		BLEU	
		NIST2006	NIST2008
(1)	baseline-NAMT (Li et al., ACL 2013)	36.3	30.0
(2)	(1) + CL(NE)-adapted-decoding	36.8	30.4
Reranking	(2) + single BiRNNLM	37.2	30.7
Adapt reranking BiRNNLM	(2) + single BiRNNLM-adapt	37.2	30.6
	(2) + CL(All)	37.4	30.9
	(2) + LDA(All)	37.0	30.7
	(2) + CL(NE)	**37.6**	**31.1**
	(2) + LDA(NE)	37.3	30.8
	(2) + single BiRNNLM + CL(NE)	**37.8**	**31.3**

Table 6. BLEU scores on the NIST2006 and NIST2008 test sets from adding sparse features, adding name-aware sparse features, and adding the best configuration of name-aware LM adaptation, compared to the baseline-NAMT. The orig inal Hiero baseline-MT BLEU scores are also listed.

	BLEU	
	NIST2006	NIST2008
baseline-MT	35.5	29.3
baseline-NAMT (Li et al., ACL 2013)	36.3	30.0
+ sparse features	37.0	30.8
++ name-aware sparse features	37.4	31.1
+++ name-aware LM adaptation	**38.4 (+2.1)**	**32.3 (+2.3)**

In our future work, we plan to investigate other name-aware sparse features, including incorporating name tagging confidences and incorporating topic information into sparse features for adaptation. We also plan to extend proposed name-aware SMT approaches to other information elements other than named entities, such as events. We will also leverage the proposed approaches on statistical machine translations for low-resource languages, where the information extraction systems for the low-resource language, e.g., name tagging, if available, could have much lower performance than the information extraction performance on languages such as Chinese.

8. ACKNOWLEDGEMENT

We thank Jing Zheng and Bing Zhao for valuable discussions. We also thank anonymous reviewers for constructive reviews.

9. REFERENCES

[1] H. Ji, R. Grishman, D. Freitag, M. Blume, J. Wang, S. Khadivi, R. Zens, and H. Ney, "Name extraction and translation for distillation," *Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitations*, 2009.

[2] Bogdan Babych and Anthony Hartley, "Improving machine translation quality with automatic named entity recognition," in *Proceedings of EAMT 2003 workshop on MT and other Language Technology Tools, Improving MT through other Language Technology Tools: Resources and Tools for Building MT*, 2003, pp. 1–8.

[3] U. Hermjakob, K. Knight, and H. Daume III, "Name translation in statistical machine translation: Learning when to transliterate," in *Proceedings of ACL*, 2008, pp. 389–397.

[4] K. Parton, K. R. McKeown, R. Coyne, M. T. Diab, R. Grishman, D. Hakkani-Tur, M. Harper, H. Ji, W. Y. Ma, A. Meyers, S. Stolbach, A. Sun, G. Tur, W. Xu, and S. Yaman, "Who, what, when, where, why? comparing multiple approaches to the cross-lingual 5w task," in *Proceedings of ACL-IJCNLP*, 2009, pp. 423–431.

[5] W. Ma and K. McKeown, "Where's the verb correcting machine translation during question answering," in *Proceedings of ACL-IJCNLP*, 2009, pp. 333–336.

[6] K. Parton and K. McKeown, "MT error detection for cross-lingual question answering," in *Proceedings of COLING*, 2010, pp. 946–954.

[7] K. Parton, N. Habash, K. McKeown, G. Iglesias, and A. de Gispert, "Can automatic post-editing make mt more meaningful?," in *Proceedings of EAMT*, 2012, pp. 111–118.

[8] Haibo Li, Jing Zheng, Heng Ji, Qi Li, and Wen Wang, "Name-aware machine translation," in *Proceedings of ACL*, 2013, pp. 604–614.

[9] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proceedings of ACL*, Philadelphia, PA, 2002.

[10] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *Proceedings of AMTA*, 2006.

[11] Mikel Artetxe, Eneko Agirre, Inaki Alegria, and Gorka Labaka, "Analyzing english-spanish named-entity enhanced machine translation," in *Proceedings of the Ninth Workshop on Syntax, Semantics and Structure in Statistical Translation*, 2015.

[12] Nora Aranberri, "SMT error analysis and mapping to syntactic, semantic and structural fixes," in *Proceedings of the Ninth Workshop on Syntax, Semantics and Structure in Statistical Translation*, 2015.

[13] David Chiang, "A hierarchical phrase-based model for statistical machine translation," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. 2005, pp. 263–270, Association for Computational Linguistics.

[14] F. J. Och and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in *Proceedings of ACL*, 2002.

[15] Stanley F. Chen and Joshua Goodman, "An empirical study of smoothing techniques for language modeling," in *Proceedings of ACL*, 1996, pp. 310–318.

[16] D. Talbot and T. Brants, "Randomized language models via perfect hash functions," in *Proceedings of ACL/HLT*, 2008, pp. 505–513.

[17] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of ACL*, 2003, pp. 160–167.

[18] Feifan Liu and Yang Liu, "Unsupervised language model adaptation incorporating named entity information," in *Proceedings of ACL*, Prague, Czech Republic, June 2007, pp. 672–679.

[19] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Machine Learning*, vol. 3, pp. 993–1022, 2003.

[20] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jam Cernocký, and Sanjeev Khudanpur, "Recurrent neural network based language model," in *Proceedings of INTERSPEECH*, 2010, pp. 1045–1048.

[21] B. Zhao and Y.-C. Tam, "Bilingual recurrent neural networks for improved statistical machine translation," in *Proceedings of SLT*, 2014, pp. 66–70.

[22] T. Mikolov, S. Kombrink, L. Burget, J. Cernocký, and S. Khudanpur, "Extensions of recurrent neural network language model," in *Proceedings of ICASSP*, 2011.

[23] David Chiang, Yuval Marton, and Philip Resnik, "Online large-margin training of syntactic and structural translation features," in *Proceedings of EMNLP*, 2008, pp. 224–233.

[24] M. Hopkins and J. May, "Tuning as reranking," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, Edinburgh, Scotland, UK, July 2011, pp. 1352–1362.

[25] B. Zhao, J. Zheng, W. Wang, and N. Scheffer, "SRI's submissions to chinese-english patentmt ntcir10 evaluation," in *Proceedings of NTCIR-10*, 2013.

[26] Q. Li, H. Li, H. Ji, W. Wang, J. Zheng, and F. Huang, "Joint bilingual name tagging for parallel corpora," in *Proceedings of CIKM*, 2012, pp. 1727–1731.

[27] Philipp Koehn, "Statistical significance tests for machine translation evaluation," in *Proceedings of EMNLP*, 2004, pp. 388–395.