A Topic-Triggered Language Model for Statistical Machine Translation

Heng Yu[†], **Jinsong Su**[*], **Yajuan lü**[†], **Qun Liu** [‡ †]
†Institute of Computing Technology, Chinese Academy of Sciences
∗ Software School of Xiamen University
‡ Centre for Next Generation Localisation
Faculty of Engineering and Computing, Dublin City University
{yuheng,lvyajuan,liuqun}@ict.ac.cn
jssu@xmu.edu.cn qliu@computing.dcu.ie

Abstract

Language model is an essential part in statistical machine translation, but traditional *n*-gram language models can only capture a limited local context in the translated sentence, thus lacking the global information for prediction. This paper describes a novel topic-triggered language model, which takes into account the topical context by estimating the *n*-gram probability under the given topics and online adjusts language model score according to different topic distributions. Experimental results show that our method provides a average improvement of +0.76 B on NIST Chinese-to-English translation task and a reduction in word perplexity of the test-document.

1 Introduction

Language model (LM) measures the fluency of translation outputs (Brown et al., 1993), and plays an important role in statistical machine translation (SMT). Traditional language model predicts the next word conditioning only on the preceding $n-1$ words, thus ignores syntactic structures in the sentence and global information over the document.

One direct approach to handle this problem is to explore sentence-level context, such as syntax-based language model for reranking (Charniak et al., 2003), and dependency language model for String-to-Dependency model (Shen et al., 2008). But these methods are still not robust enough to handle the polysemy and domain changes, as they lack the global-context information.

Another interesting line is to utilize information at document-level. Intuitively, different domains or topics have different *n*-gram probability distributions. Thus, we should take into account the topic information when we translate a document. Topic model has been learned in several parts of SMT, such as word-alignment (Zhao and Xing, 2006; Zhao and Xing, 2007; Gong et al., 2011), translation model (Xiao et al., 2012). All these works show that a particular translation often appears in some specific topical context, so it is reasonable to enhance the prediction ability of language model by incorporating topical information. Tan et al. (2011) introduces a large scale distributed composite language model incorporating document-level information. But they only focus on the target side and explore in n-best reranking task which has a limited search space, while another promising application is taking account of topical information on both sides and integrate the LM into decoding to online select translation hypotheses. However, the integration is not easy. Since the test-document can be from any topic, it is hard to dynamically estimate language model probability according to various topic distributions.

In this paper, we follow this line and introduce a novel topic-triggered language model. We first estimate the topic distribution for each document in training data, and assign those topic probabilities to each sentence. With target-side topic probabilities, we train a topic-specific language model for each topic. Then, rather than limiting topical context to target side, we utilize the source-side topical information at decoding time and online adjust language model score according to the topic distribution of the translated-document. As there is no explicit correspondence between topics on both sides, we project the source-side topic distribution to the target side as a trigger to our topic specific language models. As compared with previous works, our model takes advantage of the topical information on both sides, thus breaking down the context barrier for language model. Experimental results on various Chinese-English test sets show that our method gains an average improvement of +0.76 B points and a perplexity reduction over

International Joint Conference on Natural Language Processing, pages 447–454,
Nagoya, Japan, 14-18 October 2013.

the baseline model.

2 Related Work

Previous works devoted to improving language models in SMT mostly focus on utilizing more contextual information, such as syntax-based LMs (Charniak et al., 2003; Schwartz et al., 2011; Shen et al., 2008; Hassan et al., 2009), Forward & MI trigger LM (Xiong et al., 2011), and large-scale language models (Zhang et al., 2006; Brants et al., 2007; Emami et al., 2007; Talbot and Osborne, 2007). Since our philosophy is fundamentally different from them in that we incorporate information at document level to build language models. So we discuss previous works that explore topic information for SMT in this section.

Researchers have been trying to incorporate topic information into language models in several ways. Gildea and Hofmann (1999) use EM algorithm to perform a topic factor decomposition based on a segmented training corpus. They estimate unigram topic-based probability and combine it with standard n-gram model. Tam et al. (2007) and Ruiz and Federico (2011) introduce topic model for cross-lingual language model adaptation task. They use bilingual topic model to project latent topic distribution across languages. Based on the BLSA, they are able to transfer source-side topic weights into target-side and use them to generate topic-based marginals to adapt n-gram language model. Our model is different from theirs in that rather than using topic-based probabilities to adapt n-gram model, we directly calculate LM probability conditioned on topic distributions.

There are also some valuable applications of topic model for machine translation. Zhao and Xing (2006) propose the Bilingual Topic Admixture Model (BiTAM) for word alignment and extract topic-dependent translation model accordingly. Gong et al. (2011) introduce topic model for filtering topic-mismatched phrase pairs. Su et al. (2012) use the topic distribution of in-domain monolingual corpus to adapt the translation model. Xiao et al. (2012) introduce a topic similarity model to select the synchronous rules for hierarchical phrase-based translation. Our work is in the same spirit with those works, but we are interested in LM problem rather than other parts in SMT.

Our work models topic probabilities into training corpus and trains several topic-specific LMs, so it is in the same spirit of mixture modeling. Heidel et al. (2007) use topic distribution to cluster the training corpora and train LMs accordingly. Our method is different from theirs in that we assign topic probabilities to training sentences rather than segment them into different topics, so our model is more robust to data sparse problem. Besides, Foster and Kuhn (2007), Civera and Juan (2007), Lü et al. (2007) also adapt mixture modeling framework to exploit the full potential of existing corpus. Adopting this framework, the training corpus is first divided into different parts, each of which is used to train a sub model, then these sub models are used together with different weights during decoding. Those works typically use word similarities and sentence level information, while our work extents the context into the document level.

3 Topic triggered Language Model

Polysemy is a difficult problem for statistical machine translation. As shown in Figure 1, English sentence "give me a shot" has different meanings in different domains. Using traditional LM, which only considers the local context information in the translated sentence, this ambiguous translation is hard to handle, since these translations are all common in the corpus with different domains. But with the help of topical context information, the difference can be told. For example, the word 'shot' is often translated into "(photo)" in the sentences related to the film topic, and to "(chance)" in sports topic. So as the topic information is concerned, LM allows for more fine-grained distinction of different translations and enjoys stronger prediction power.

In our method, we introduce the topic of current document t as a hidden variable, and decompose the language model probability as follows:

$$\mathrm{P}(\mathbf{e}) = \sum_{t} \mathrm{P}(\mathbf{e}, t) = \sum_{t} \mathrm{P}(\mathbf{e}|t) \cdot \mathrm{P}(t) \qquad (1)$$

$\mathrm{P}(\mathbf{e}|t)$ indicates the probability of the sequence $\mathbf{e}$ given the topic t, and $\mathrm{P}(t)$ is the topic distribution of the test-document which is calculated during decoding. In general, our framework to build the topic-trigger language model can be specified into two steps:

- Build topic-specific LMs conditioned on the topic distribution estimated by the target-side topic model

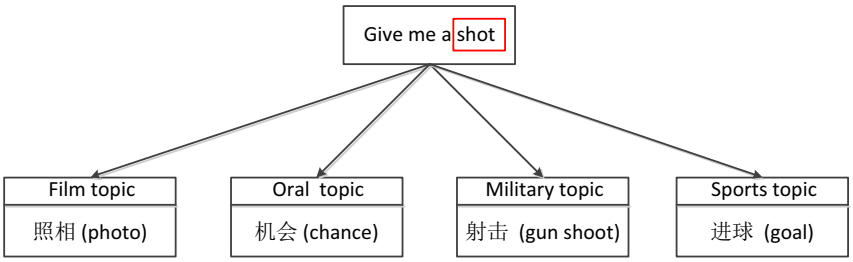

Figure 1: Example of different translations of word "shot" in different topics

- Capture source-side topic information during decoding and online adjust LM score

We will give detailed description of the two parts in the following section.

3.1 Topic-specific language model

In this section, we first briefly review the principle of Hidden Topic Markov Model (HTMM) which is the basis of our method, then describe our approach to build topic-specific LMs in detail.

3.1.1 Hidden Topic Markov Model

Topic model is a suite of algorithms aiming to discover the hidden thematic structure in large archives of documents. Recently both Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and Probabilistic Latent Semantic Analysis (PLSA) (Hofmann, 1999) have been successfully applied in various NLP tasks. Based on the "bag-of-words" assumption that the order of words can be ignored, these methods model the corpus as a co-occurrence matrix of words and documents, and build generative models to infer the latent aspect of topics. Using these models, words can be clustered into the derived topics with a probability distribution. and the correlation between words can be automatically captured via topics.

However, the "bag-of-words" assumption is an unrealistic oversimplification in language model case because it ignores the order of words which is critical in estimating n-gram probabilities. To remedy this problem, we use Hidden Topic Markov Models (HTMM), proposed by (Gruber et al., 2007) , which models the topics of words in the document as a Markov chain. The model is based on the assumption that all words in the same sentence share the same topic and the successive sentences are more likely to have the same topic. HTMM incorporates the local dependency between words by Hidden Markov Model for better topic estimation.

3.1.2 Topic Probability Assignment

We use HTMM (Gruber et al., 2007) to train topic model on our training set and obtain sentence-level topic probabilities. To avoid data sparse problems, we use the topic probability of each sentence as a soft clustering for each topic rather than force hard decisions on topic assignment. In this way, we are able to get n-gram distributions for different topics. So the topic-sensitive words will have a higher occurrence in specific topics while common words will distribute uniformly in every topic.

3.1.3 Estimation

We follow the common practise in n-gram model (Goodman, 2001) and simplify $P(\mathbf{e}|t)$ into a serial of n-gram probabilities $\mathrm{P}(w_i|w_{i-n+1}^{i}, t)$ based on Markov Assumption. Formally, we decompose the probability as follows:

$$\mathrm{P}(\mathbf{e}|t) = \mathrm{P}(w_1|t)\cdot\mathrm{P}(w_2|w_1,t)\cdots\mathrm{P}(w_i|w_{i-n+1}^{i},t) \quad (2)$$

Noted that, based on HTMM, we assume that all words in one sentence share the same topic, so topic t in Equation 2 can be shared. To compute $\mathrm{P}(w_i|w_{i-n+1}^{i}, t)$, We use Maximum-Likelihood Estimation (MLE) with the n-gram fractional count for each topic. And since some topic-based n-grams probabilities are sharply distributed, we use Witten-Bell(WB) method (Witten and Bell, 1991) for smoothing.

$$\mathrm{P}_{MLE}(w_i|w_{i-n+1}^{i-1}, t_e) = \frac{Count(w_{i-n+1}^{i}, t_e)}{Count(w_{i-n+1}^{i-1}, t_e)} \quad (3)$$

$$\begin{aligned}\mathrm{P}(w_i|w_{w_{i-n+1}}^{i-1}, t_e) = \lambda_{w_{i-n+1}^{i-1}}\mathrm{P}_{MLE}(w_i|w_{w_{i-n+1}}^{i-1}, t_e) \\ +(1-\lambda_{w_{i-n+1}^{i-1}})\mathrm{P}(w_i|w_{w_{i-n+2}}^{i-1}, t_e)\end{aligned} \quad (4)$$

In Equation 4, λ is a normalization parameter for MLE probability and back-off probability,

which can be calculated using the following equation:

$$\lambda_{w_{i-n+1}^{i-1}} = \frac{N_{1+}(w_{i-n-1}^{i-1}, t_e)}{N_{1+}(w_{i-n-1}^{i-1}, t_e) + \sum_{w_i} c(w_{i-n+1}^{i}, t_e)} \quad (5)$$

where $N_{1+}(w_{i-n-1}^{i-1}, t_e)$ denotes for the number of words w following w_{i-n-1}^{i-1} in topic t_e, and $c(w_{i-n+1}^{i}, t_e)$ is the count of n-gram w_{i-n+1}^{i} in t_e.

3.2 Integration with SMT

We integrate our LM into SMT system to utilize topic distribution of the test-document as a trigger to each topic-specific language model. But as we know, only source side is available before decoding in SMT. So in order to get target-side topic distribution $P(t_e)$, we need to estimate the source-side topic distribution $P(t_f)$ and then project it to the target side. So Equation 1 can be further refined as the following Equation:

$$\mathrm{P}(\mathbf{e}) = \sum_{t_e} \mathrm{P}(\mathbf{e}|t_e) \cdot \sum_{t_f} \mathrm{P}(t_e|t_f) \cdot \mathrm{P}(t_f) \quad (6)$$

where $\mathrm{P}(t_e|t_f)$ is the topic projection probability.

3.2.1 Topic Projection

Since topic distributions of bilingual sentences often share the same pattern (Gao et al., 2011), we follow the work of Xiao et al. (2012) and introduce the topic projection probability $P(t_e|t_f)$ to project the source-side topic distribution into the target-side topic space. We train topic models on both sides of the training data, then with the help of the word alignment we estimate the projection probability by the co-occurrence of the source-side and the target-side topic assignment.

Formally, we denote each parallel sentence pair by $(t_f, t_e, \mathbf{a})$, where t_f and t_e are the topic assignments of source-side and target-side sentences respectively, and $\mathbf{a}$ is a set of word alignments $\{(f_i, e_j)\}$. An alignment (i, j) denotes source-side word f_i aligns to target-side word e_j, so the topics of both words are also aligned. Thus, the co-occurrence of a source-side topic with index d_f and a target-side topic d_e, $Cnt(t_f, t_e)$ is calculated by:

$$Cnt(t_f, t_e) = \sum_{(t_f, t_e, \mathbf{a})} \sum_{(i,j)\in\mathbf{a}} \delta(t_{f_i}, d_f) * \delta(t_{e_j}, d_e) \quad (7)$$

where $\delta(x; y)$ is the Kronecker function, which is 1 if x = y and 0 otherwise. We then compute the probability of $\mathrm{P}(t = d_f, t = d_e)$ by normalizing the co-occurrence count. Overall, we obtain a correspondence matrix $M_{d_e \times d_f}$ from target-side topic to source-side topic, where the item $M_{i;j}$ represents the probability $\mathrm{P}(t_f = i, t_e = j)$. Then with the correspondence matrix $M_{d_e \times d_f}$, we are able to project the source-side topic $P(t_f)$ to the target-side topic space, which we called projected target-side topic distribution $T(\mathrm{P}(t_f))$.

3.2.2 Topic-triggered Estimation

During decoding, we first estimate the source-side topic distribution of the test-set $P(t_f)$, then using the topic projection matrix, we map $P(t_f)$ to the target side, and generate each topic t_e with probability $P(t_e|t_f)$. Then topic t_e triggers its topic-specific LM $P(e|t_e)$. We use the weighted sum of each model as the final LM score.

4 Experiments and Results

We try to answer the following questions by experiments:

- Can our topic-triggered language model help improve translation quality in terms of both B and perplexity.
- How is the topic number affect the language model performance.
- Can our model make better use of training corpus than N-gram model.

4.1 Experiment setup

We present our experiments on the NIST Chinese-English translation tasks. The bilingual training data for translation model contain 1.5M sentence pair with 38M Chinese words and 32M English words. The monolingual data for training English language model includes the Xinhua portion of the GIGAWORD corpus, which contains 10M sentences. We used the NIST evaluation set of 2006(MT06) as our development set, and sets of MT04/05/08 as test sets. Corpus statistics are shown in Table 1.

We obtain symmetric word alignments of training data by first running GIZA++ (Och and Ney, 2004) in both directions and then applying refinement rule "grow-diag-final-and" (Koehn et al., 2003). We re-implement the Hierarchical phrase-based system (Chiang, 2007) and extract SCFG

Data	Sentence	documents
Language model training	10M	980K
Translation model training	1.5M	99.4K
Tuning	616	52
Testing(04)	1788	200
Testing(05)	1082	100
Testing(08)	1357	109

Table 1: Training, tuning and test data used for evaluating B score.

ppl	04	05	08
Base LM	158.42	134.59	208.11
Topic LM	148.11	119.17	200.41

Table 3: 4-gram word perplexity results of our method in terms of *ppl*, We compare our model with baseline *n*-gram model ("Base LM") on three test-sets.

rules from this word-aligned training data. A 4-gram language model is trained on the monolingual data by SRILM toolkit (Stolcke, 2002). Case-insensitive NIST BLEU (Papineni et al., 2002) is used to measure translation performance. We use minimum error rate training (Och, 2003) for optimizing the feature weights.

To obtain topic distribution, We use the open source LDA tool Open HTMM developed by Gruber et al. (2007) for estimation and inference. During this process, we empirically set the parameter values for HTMM training as: $\alpha = 1.5, \beta = 1.01, iters = 100$. See Gruber et al. (2007) for the meanings of these parameters. and set the topic number to 30 [1] for both source and target side. The source-side topic model is estimated from the Chinese part of training corpus, while the target side is estimated from both Xinhua and the English side of training corpus.

4.2 Effect of topic-trigger language model

For machine translation task, our baseline is the traditional hiero system with standard features (Chiang, 2007). The baseline language model is a 4-gram model trained on Xinhua corpus. Noted that we use Keneser-Ney smoothing (Kneser and Ney, 1995) for baseline LM since it's universally acknowledged to achieve better performance. And our topic-triggered language model is trained on the same corpus with topic distribution estimated from topic model. We add our model as an new feature into the system, denote as STLM. To prove the soundness of our approach, we re-implement two comparative experiments: HTLM makes hard-decision on topic selection in both training and decoding, assigning the topic with the highest probability to the sentence, which is in the same spirit with the Heidel et al. (2007) method. Second, we follow the method by Tam et al. (2007), denote as "Tam", and generate topic-based marginals to adapt *n*-gram language model.

Table 2 reports the B and TER scores on all test-sets. The baseline system achieves B score of 37.43 on NIST04, 33.67 on NIST05 and 28.54 on NIST08 set. Our method(STLM) gains an average improvement of +0.76 B and an average reduction of −0.88 TER over the baseline. Results on NIST MT 04, 05, 08 are statistically significant with $p < 0.05$ (Koehn, 2004). This verifies that our topic-triggered language model is a good complement for *n*-gram model to further improve translation quality. We can also see that our method generally out-performs the Tam's method, because our model can capture n-gram level topic information, rather than only focus on estimating 1-gram topic-based probability. Another interesting result is forcing hard-decision on topic selection (HTLM) only achieves a little improvement over the baseline. The reason is two folded: First, in LM training process, the hard-decision on topic will serve as a corpus split strategy and cause data sparse problems. Second in decoding, one sentence may not solely belong to one topic, so the hard decision will cause inaccurateness in LM prediction.

We then evaluate our method in terms of perplexity. As an initial measure to compare language models, average per-word perplexity(*ppl*) , reports how surprised a model is by test data. Equation 8 calculates *ppl* using log base b for a test set of T tokens.

$$ppl = b^{\frac{-\log_b \mathrm{P}(e_1 \ldots e_T)}{T}} \quad (8)$$

we evaluate 4-gram perplexity of the translation hypotheses using baseline language model and our topic-triggered model.

Table 3 shows that our model reduces the average word perplexity by 6% compared to baseline language model. The results indicate that our model successfully leverages the source-side document and reduces the *ppl* on the target side.

[1] We determine the topic number by testing 5, 10, 15, 30, 50 in our preliminary experiments. We find that 30 topics produces a slightly better performance than other values.

Model	04		05		08		AVG	
	B	TER	B	TER	B	TER	B	TER
Baseline	37.43	39.88	33.67	42.45	28.54	47.32	33.21	43.22
Tam	37.86	39.12	34.28	41.93	29.02	46.92	33.72	42.66
HTLM	37.46	39.86	33.74	42.41	28.67	47.23	33.29	43.17
STLM	**38.28**	**38.95**	**34.30**	**41.93**	**29.32**	**46.14**	33.97	42.34

Table 2: Results of our method in terms of B / TER, "Tam" dentes using topic adaptation method from Tam et al. (2007). "HTLM" denotes using topic-triggered LM with hard decision of topic assignment, and "STLM" means topic assignment by probabilities. Scores marked in bold are statistically significantly with $p < 0.05$ (Koehn, 2004).

Test-set	04	05	08
baseline	31.31	28.43	23.67
5 topics	30.81	27.96	23.26
10 topics	30.98	28.12	23.42
15 topics	31.32	28.40	23.64
20 topics	31.39	28.40	23.82
30 topics	31.75	28.51	24.05
50 topics	31.70	28.48	24.01

Table 4: Results on all test sets with different topic number.

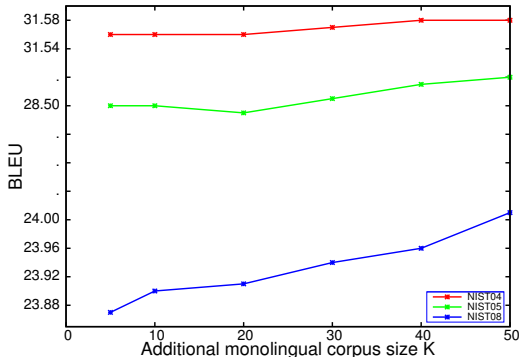

Figure 2: BLEU improvement with additional topic-modeling training corpus

4.3 Effect of topic number

In topic model training, topic number is a manually set parameter. However, as an empirical factor, the topic number diverse a lot in different training corpus. so it's worthy to explore the effect of topic number on the performance of our topic-trigger language model. We set topic number to 5, 10, 15, 20, 30, 50 respectively to train topic models on both sides.

Table 4 shows the B scores using 5, 10, 15, 20 ,30, 50 topics. We can see that with only 5 topics, the model performance is a little worse than the baseline model. This is reasonable because the corpus has not been fully clustered into different topics, so the topic information has not been fully utilized. But we can see ,as the topic number grows larger, the performance gets better with a peak at 30 topics, resulting a 0.34 improvement average over the baseline.

But there is a little slump when it comes to 50, we think the reason is as we models topic distribution into the LM training corpus, the distribution gets too scattered as topic number grows causing data-sparse problem in topic-specific language model training, thus affecting the overall probability of the language model.

4.4 Effect of better topic model estimation

Finally, we investigate the effect of larger topic-training corpus. One important feature of topic model is the larger the training corpus is , the better model we will get. In our experiment, we use the source of fbis which only have 10,947 documents to train source-side topic model. This may not be good enough to correctly estimate the topic distribution of the test set, since we know that NIST08 contains a large portion of web corpus. So we add different size of source-side monolingual corpus: 5K, 10K, 20K, 30K, 40K, 50K from Chinese Sohu weblog corpus[2] only to train different source-side topic models with 30 topics.

Figure 4 shows the B scores of the translation system on NIST04,05,08. It can be seen that additional corpus improves translations quality on NIST08. This is because the additional corpus expand the diversity of the topic model, especially for NIST08 which contains a large part of web data, generating more accurate topic distribution. The best B comes to 24.12 when the additional corpus size is 50K, achieving 0.42 gains on the

[2]http://blog.sohu.com

baseline system. But on 04 and 05 test-sets, the improvement is not that significant. This may because the 04, 05 set are not similar with the additional corpus, so they are not effected by the improvement of topic model. The results indicates that the performance of our topic-triggered language model is directly associated with the topic model, which can be improved by training with larger and more relative corpus.

5 Conclusion

In this paper. we follow this line and introduce a novel topic-triggered LM. We first estimate the topic distribution for each document in training data, and assign those topic probabilities to each sentence, then, we train a topic-specific *n*-gram LM for each topic based on those topic probabilities. At decoding time, as target translations are not available before translation, we simply project the topic distribution from source to target side. Then we compute the topic-triggered LM score according to the topic distribution of the translated-document. Experimental results show that our model achieves better performance than traditional *n*-gram model on both perplexity and B score.

In the future, we will verify our method in other domain and language pairs. Further more, we want to combine our work with other related works to see if it can further improve the translation quality. Finally, we will explore more robust framework to incorporate syntax and semantic information to make our language model more powerful.

6 Acknowledgments

The authors were supported by 863 State Key Project (No. 2011AA01A207), and National Key Technology R&D Program (No. 2012BAH39B03). Jinsong Su's work is supported by Reseach Fund for the Doctoral Program of Higher Education of China (Grant NO. 20120121120046). Qun Liu's work is partially supported by Science Foundation Ireland (Grant No.07/CE/I1142) as part of the CNGL at Dublin City University. We would like to thank the anonymous reviewers for their insightful comments and those who helped to modify the paper.

References

David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. 2003. Latent dirichlet allocation. In *Journal of Machine Learning Research*, volume 3, page 2003.

Thorsten Brants, Ashok C. Popat, Peng Xu, Franz J. Och, Jeffrey Dean, and Google Inc. 2007. Large language models in machine translation. In *In EMNLP*, pages 858–867.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311, June.

Eugene Charniak, Kevin Knight, and Kenji Yamada. 2003. Syntax-based language models for statistical machine translation. In *In MT Summit IX. Intl. Assoc. for Machine Translation.*

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33.

Jorge Civera and Alfons Juan. 2007. Domain adaptation in statistical machine translation with mixture modelling. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

A. Emami, K. Papineni, and J. Sorensen. 2007. Large-scale distributed language modeling. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–37 –IV–40, april.

George Foster and Roland Kuhn. 2007. Mixture-model adaptation for smt. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 128–135, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jianfeng Gao, Kristina Toutanova, and Wen-tau Yih. 2011. Clickthrough-based latent semantic models for web search. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 675–684, New York, NY, USA. ACM.

Daniel Gildea and Thomas Hofmann. 1999. Topic-based language models using em. In *In Proceedings of EUROSPEECH*, pages 2167–2170.

Z. Gong, G. Zhou, and L. Li. 2011. Improve smt with source-side "topic-document" distributions. In *Machine Translation Summit XIII*, page 496.

Joshua T. Goodman. 2001. A bit of progress in language modeling. In *Technical Report MSR-TR-2001-72*.

Amit Gruber, Yair Weiss, and Michal Rosen-Zvi. 2007. Hidden topic markov models. In *International Conference on Artificial Intelligence and Statistics*, pages 163–170.

Hany Hassan, Khalil Sima'an, and Andy Way. 2009. A syntactified direct translation model with linear-time decoding. In *Proceedings of EMNLP 2009*, pages 1182–1191, Singapore, August. Association for Computational Linguistics.

Aaron Heidel, Hung an Chang, and Lin-Shan Lee. 2007. Language model adaptation using latent dirichlet allocation and an efficient topic inference algorithm. pages 2361–2364.

T. Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM.

R. Kneser and H. Ney. 1995. Improved backing-off for m-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 1, pages 181 – 184 vol.1, may.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. of HLT-NAACL 2003*, pages 127–133.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proc. of EMNLP 2004*, pages 388–395.

Yajuan Lü, Jin Huang, and Qun Liu. 2007. Improving statistical machine translation performance by training data selection and optimization. In *EMNLP-CoNLL*, pages 343–350.

Franz Joseph Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, pages 417–449.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL 2003*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL 2002*, pages 311–318.

Nick Ruiz and Marcello Federico. 2011. Topic adaptation for lecture translation through bilingual latent semantic models. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT '11, pages 294–302, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lane Schwartz, Chris Callison-Burch, William Schuler, and Stephen Wu. 2011. Incremental syntactic language models for phrase-based translation. In *Proceedings of ACL 2011*, pages 620–631, June.

Libin Shen, Jinxi Xu, and Ralph M Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *ACL*, pages 577–585.

Andreas Stolcke. 2002. Srilm - an extensible language modeling toolkit. In *Proc. of ICSLP 2002*, pages 901–904.

Jinsong Su, Hua Wu, Haifeng Wang, Yidong Chen, Xiaodong Shi, Huailin Dong, and Qun Liu. 2012. Translation model adaptation for statistical machine translation with monolingual topic information. In *Proceedings of ACL 2012*, pages 459–468, Jeju Island, Korea, July. Association for Computational Linguistics.

David Talbot and Miles Osborne. 2007. Randomised language modelling for statistical machine translation. In *Prague, Czech Republic. Association for Computational Linguistics*, pages 512–519.

Yik-Cheung Tam, Ian Lane, and Tanja Schultz. 2007. Bilingual-lsa based lm adaptation for spoken language translation. In *Proceedings of ACL 2007*, pages 520–527, Prague, Czech Republic, June. Association for Computational Linguistics.

Ming Tan, Wenli Zhou, Lei Zheng, and Shaojun Wang. 2011. A large scale distributed syntactic, semantic and lexical language model for machine translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 201–210, Portland, Oregon, USA, June. Association for Computational Linguistics.

Ian H Witten and Timothy C Bell. 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *Information Theory, IEEE Transactions on*, 37(4):1085–1094.

Xinyan Xiao, Deyi Xiong, Min Zhang, Qun Liu, and Shouxun Lin. 2012. A topic similarity model for hierarchical phrase-based translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 750–758, Jeju Island, Korea, July. Association for Computational Linguistics.

Deyi Xiong, Min Zhang, and Haizhou Li. 2011. Enhancing language models in statistical machine translation with backward n-grams and mutual information triggers. In *ACL*, pages 1288–1297.

Ying Zhang, Almut Silja, and Hildebrand Stephan Vogel. 2006. Distributed language modeling for n-best list re-ranking. In *in Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 216–223.

B. Zhao and E.P. Xing. 2006. Bitam: Bilingual topic admixture models for word alignment. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 969–976. Association for Computational Linguistics.

Bing Zhao and Eric P Xing. 2007. Hm-bitam: Bilingual topic exploration, word alignment, and translation. In *Advances in Neural Information Processing Systems*, pages 1689–1696.