



Using Features from Topic Models to Alleviate Over-generation in Hierarchical Phrase-based Translation

Songfang Huang, Bowen Zhou

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598, USA

{shuang, zhou}@us.ibm.com

Abstract

In hierarchical phrase-based translation systems, the grammars (SCFG rules) have over-generation problem because we can replace the non-terminal X with almost everything without knowing the syntactic or semantic role of X . In this paper, we present an approach that uses topic models to learn the distributions for non-terminals in each SCFG rule, based on which we further derive static features for the discriminative framework of statistical machine translation. Experimental results on three corpora show that we can obtain some gains in BLEU by using these features derived from topic models to alleviate the over-generation problem in hierarchical phrase-based translation.

Index Terms: hierarchical phrase-based translation, topic model, feature, over-generation

1. Introduction

In recent years, there are growing research interests in using syntax for statistical machine translation (SMT). Syntax-based translation systems have shown better translation performance than phrase-based systems [1]. One example is the hierarchical phrase-based translation model in [2], which uses the synchronous context-free grammar (SCFG) to automatically extract hierarchical structures of natural language.

The hierarchical phrase-based translation [2] is also referred to *formally* syntax-based model because the hierarchical syntax structures are inferred without any explicit linguistic knowledge or annotations. On the other hand, another group of models that relies on linguistic knowledge (i.e., from a parser) or linguistic annotation (i.e., from Penn Treebank) is called *linguistically* syntax-based translation. Examples include, but not limited to, [3, 4, 5].

An SCFG is a synchronous rewriting system generating source and target side string pairs simultaneously based on a context-free grammar. Each synchronous production (i.e., rule) rewrites a non-terminal into a pair of strings, γ and α , with both terminals and non-terminals in both languages. In particular, formal syntax-based models explore hierarchical structures of natural language and utilize only a unified non-terminal symbol X in the grammar:

$$X \rightarrow \langle \gamma, \alpha, \sim \rangle \quad (1)$$

where \sim is the one-to-one correspondence between X 's in γ and α .

One problem of SCFG rules in hierarchical phrase-based translation is that those rules with non-terminals X could be over-generation, in the sense that X could be replaced by almost everything without knowing the syntactic or semantic role

of X . Some previous studies, therefore, have incorporated linguistic knowledge into formally syntax-based models, to constraint the over-generation of non-terminals and obtain compensated and augmented performance. For example, [6] used a hard constraint approach by directly using explicit treebank categories to annotate phrase pairs. [7], instead, used a soft constraint method, by inducing a set of latent categories to capture the syntactic dependencies, and deriving a real-valued feature vector for each X non-terminal of an SCFG rule based on the distribution of the latent categories. However, one disadvantage of this method is that it highly relies on a parser which could be inefficient during decoding. A workshop¹ at John Hopkins University (JHU) also addressed this over-generation problem by classifying the non-terminal X into sub-categories, i.e., X_1, X_2, \dots, X_N , according to the classification of neighbouring contexts. The categories encode the syntactic role of the phrase pairs. A nonparametric Bayesian model based on hierarchical Pitman-Yor processes [8] was used for the unsupervised category induction.

Similar to [7] and the work at the JHU workshop, we try to alleviate the over-generation of non-terminals X 's in SCFG rules. The idea is that we learn distributions for non-terminals in SCFG rules from topic models, which encodes the semantic role of what X represents given a training corpus. This method has some advantages comparing to previous work. Unlike [7] which set some soft constraints using knowledge from a parser, we in this paper avoid the usage of a parser, but use topic models to induce the latent categories (topics). Unlike the work at the JHU workshop which largely increases the models size and consequently slows down the decoding by using more categories for X , we differentiate the semantic roles of X using the classification property of topic models over the latent topic simplex. Another difference is that we directly classify the phrase pairs that X represents given a training corpus using topic models, rather than the neighbouring contexts in the JHU workshop. Finally, this work addresses the over-generation problem of X for SCFG rules at the semantic level.

2. Probabilistic Topic Models

Topic models, which have received a growing interest in the machine learning community, aim to find a latent representation connecting documents and words — the topic. In a topic model, words in a document exchangeably co-occur with each other according to their semantics, following the “bag-of-words” assumption.

Suppose there are D documents in the corpus, and W

¹<http://www.cisp.jhu.edu/workshops/ws10/groups/msgismt/>

words in the vocabulary. Each document $d = 1, \dots, D$ in the corpus is represented as a mixture of latent topics, with the mixing proportions over topics denoted by θ_d . Each topic $k = 1, \dots, K$ in turn is a multinomial distribution over words in the vocabulary, with the vector of probabilities for words in topic k denoted by ϕ_k .

Latent Dirichlet allocation (LDA) [9] is a three-level hierarchical Bayesian model, which pioneered the use of the Dirichlet distribution for latent topics. That is, the topic mixture weights θ_d for the d th document are drawn from a prior Dirichlet distribution with parameters α, π :

$$P(\theta_d | \alpha \pi) = \frac{\Gamma(\sum_{i=1}^K \alpha \pi_i)}{\prod_{i=1}^K \Gamma(\alpha \pi_i)} \theta_1^{\alpha \pi_1 - 1} \dots \theta_K^{\alpha \pi_K - 1} \quad (2)$$

where K is the predefined number of topics in LDA, Γ is the Gamma function, $\alpha \pi = \{\alpha \pi_1, \dots, \alpha \pi_K\}$ represents the prior observation counts of the K latent topics with $\alpha \pi_i > 0$: π is the corpus-wide distribution over topics, and α is called the concentration parameter which controls the amount of variability from θ_d to their prior mean π .

Similarly, Dirichlet priors are placed over the parameters ϕ_k with the parameters $\beta \tau$. We write:

$$\theta_d | \pi \sim \text{Dir}(\alpha \pi) \quad (3)$$

$$\phi_k | \tau \sim \text{Dir}(\beta \tau) \quad (4)$$

The generative process for words in each document is as follows: first draw a topic k with probability θ_{dk} , then draw a word w with probability ϕ_{kw} . Let w_{id} be the i th word token in document d , and z_{id} the corresponding drawn topic, then we have the following multinomial distributions:

$$z_{id} | \theta_d \sim \text{Mult}(\theta_d) \quad (5)$$

$$w_{id} | z_{id}, \phi_{z_{id}} \sim \text{Mult}(\phi_{z_{id}}) \quad (6)$$

the main objectives of inference in LDA are to find (1) the word distribution $P(\mathbf{w} | z_i, \beta)$ for each topic z_i , and (2) the topic distribution $P(\theta | \alpha)$ for each document. Since exact inference for the posterior distributions in LDA is intractable, a wide variety of approximate inference algorithms can be used for LDA, including variational methods [9], and Markov-chain Monte Carlo (MCMC) methods such as Gibbs sampling [10] and collapsed Gibbs sampling [11].

Hierarchical Dirichlet process (HDP) [12] is a Bayesian nonparametric model that is useful to model multiple groups of data in which each group of data is associated with an underlying parameter and these parameters are shared together across groups. By using a nonparametric prior called Dirichlet process [13], the HDP is capable of automatically inferring the number of topics from the data.

3. Topic Modeling for SCFG Rules

The process for automatically extracting SCFG rules from training data in a hierarchical phrase-based translation is as follows [2]. First we extract bilingual phrase pairs, denoted as \mathcal{BP} , from the union of bidirectional word-level alignments [14]. Each phrase pair $\text{PP} \in \mathcal{BP}$ is represented as a production rule $X \rightarrow \langle f_i^j, e_k^l \rangle$. Next, we loop through each phrase pair PP and generalize the sub-phrase pair contained in PP , denoted as SP_e and SP_f subject to $\text{SP} = (\text{SP}_f, \text{SP}_e) \in \mathcal{BP}$, with co-indexed non-terminal symbols X . We thereby obtain a new rule.

During rule extraction, we recursively replace sub-phrase pairs to obtain SCFG rules. For a rule R , it can be extracted

from different bilingual phrase pairs. This means the non-terminal X in a given rule R can represent different but limited word sequences based on a given training corpus. During decoding, however, we apply the rule R without knowing the syntactic or semantic role of the word sequence that X represents. To solve this mismatch problem, we use topic models to learn the information about what X of a rule R can represent given a training corpus in a probabilistic way. The detailed steps are as follows:

1. During rule extraction, we collect the sub-phrases that represented/replaced by non-terminal X from each bilingual phrase pair, and put these sub-phrases together as a ‘‘bag-of-words’’ document for each non-terminal rule. For rules with two non-terminals, we collapse the two word sequences, one for each non-terminal. Each distinct rule has a corresponding document.
2. We apply topic models over these exchangeable documents and carry out the inference. In this paper, we use latent Dirichlet allocation [9] for topic modeling.
3. After convergence, we obtain a multinomial distribution $\theta^r = \{\theta_1^r, \dots, \theta_K^r\}$ over K topics for each distinct rule r . Each topic $\phi_k = \{\phi_{kw_1}, \dots, \phi_{kw_n}\}$ in turn is a multinomial distribution over the vocabulary. The K topics are shared by all rules.

4. Using as Static Features

We describe how we derive static features for the source side, denoted as F_S , based on the distributions. A rule with the same source side E^r may come from different phrase pairs PP_1^r, \dots, PP_m^r , by replacing word sequences s_1^r, \dots, s_m^r as a non-terminal X in the source side respectively. We denote the instances of these unmerged rules as r_1^E, \dots, r_m^E . They have corresponding distributions $\{\theta_r, \phi\}$ from topic models to represent X . We can therefore calculate a probability of how likely each $s_i^r \in \{s_1^r, \dots, s_m^r\}$ is generated by the distributions in $r_i^E \in r_1^E, \dots, r_m^E$:

$$P(s_i^r | \theta_r, \phi) = \prod_{j=1}^J \sum_{k=1}^K \theta_k^r \cdot \phi_{kw_j} / J \quad (7)$$

where J is the number of words in s_i^r . We treat $P(s_i^r | \theta_r, \phi)$ as a pseudo-count for r_i^E . By normalizing these pseudo-counts according to different target side of r_i^E , we can obtain features based on the source side for each distinct rule. For phrasal rules, we simply set the features to a fixed value, e.g., 1.0.

Intuitively, the idea is that we re-evaluate how good it is to extract a candidate rule from a phrase pair during rule extraction, by globally considering what X in the candidate rule can represent.

The similar approach can be applied to obtain features from the target side, denoted as F_T .

Translation using SCFG grammars for an input sentence E is casted as to find the optimal derivation on the source and target sides (as the grammar is synchronous, the derivations on source and target sides are identical). That is, the goal of decoding is to search for the best derivation D that maximizes the following log-linear model over all possible derivations:

$$P(D) \propto P_{LM}(e)^{\lambda_{LM}} \times \prod_i \prod_{X \rightarrow \langle \gamma, \alpha \rangle \in D} \phi_i(X \rightarrow \langle \gamma, \alpha \rangle)^{\lambda_i}, \quad (8)$$

where the set of $\phi_i(X \rightarrow \langle \gamma, \alpha \rangle)$ are features defined over given production rule, and $P_{LM}(e)$ is the language model score on hypothesized output, the λ_i is the feature weight. A bottom-up CKY parser is widely used for decoding.

We can easily incorporate the features from topic models in the discriminative translation framework in Equation 8.

5. Experiments and Results

We conduct experiments on three corpora: Chinese-to-English on IWSLT 2006, English-to-Chinese on internal data, and English-to-German on Europarl. The statistics of the corpora is shown in Table 1.

Table 1: The statistics of the three corpora.

Task		#Train	#Dev	#Test	#Ref
IWSLT	F2E	39,953	489	500	7
Chinese	E2F	482,017	1,378	1,377	1
Europarl	E2F	296,999	1,000	1,000	2

The baseline system we used in this paper is a state-of-the-art formally syntax-based translation system, as described in detail in [15]. The baseline system has seven basic features including four rule and lexicalized translation probabilities plus three discounting features. We used 4-gram language models with modified Kneser-Ney smoothing [16] for all the three tasks. Minimum-error-rate training (MERT) [17] was carried out to optimize the feature weights on the dev set.

We used the implementation by [9]² for topic modeling. We initialized the LDA models with 30 topics for IWSLT data, and with 40 topics for both the internal Chinese corpus and the EUROPARL corpus. The inference scheme is variational inference. The hyperparameters are automatically estimated along with the topic distributions.

We evaluated the effect of features from the source side (F_S) and the target side (F_T) respectively for each of the three corpora.

Table 2 shows the results in BLEU [18] on the three corpora. We observe some gains in BLEU using the additional static features from topic models. We note that the proposed method for alleviating the over-generation problem in this paper is not resource intensive — we append only static features for each SCFG rules which does not significantly increase the decoding time. We can combine the features with the tuned weights, and store one single pre-computed feature value for each SCFG rule in the model.

Table 2: The BLEU results. The results on Europarl are based on models without monotone rules. The bracket indicates if the features are from the source side (F_S) or the target side (F_T).

Task		Baseline	+Feature	Δ
IWSLT	Dev	21.61	21.86 (F_T)	0.25
	Test	21.74	22.45 (F_T)	0.71
Chinese	Dev	46.88	47.34 (F_S)	0.46
	Test	45.42	45.88 (F_S)	0.46
Europarl	Dev	16.50	16.81 (F_T)	0.31
	Test	15.68	16.49 (F_T)	0.81

²<http://www.cs.princeton.edu/~blei/lda-c/index.html>

6. Discussion

In this paper we apply topic models over short documents, i.e., phrases. Basically, we can regard this as a classification problem for phrases on the continuous topical space. We found in the experiments that we can obtain reasonable topics in this way. Table 3 illustrates five example topics from the Chinese corpus, each showing top 30 words with the corresponding probabilities in the topic.

We also experimentally verified the effect of topic numbers in LDA. It seems that topic numbers we set in the experiments are reasonable — changing the number of topics within small range does not change the translation performance much. The HDP [12] claims the ability to automatically induce the number of topics from the data. We used the implementation of the HDP by [19]³, and confirmed that the topic numbers we set basically match the induced numbers by the HDP.

There are two interesting follow-up work. Firstly, we can use as dynamic features instead of static features introduced in Section 4. Each distinct rule is associated with distributions $\{\theta_r, \phi\}$. Using the distributions based on the source side F_S , we can dynamically impose soft constraints during decoding. Suppose we apply a rule r to an input span with word sequence $w_i^j = \{w_i, \dots, w_j\}$. Similar to Equation 7, we can compute on the fly a feature for r as follows:

$$P(w_i^j | \theta_r, \phi) = \prod_{n=i}^j \sum_{k=1}^K \theta_k^r \cdot \phi_{kw_n} \quad (9)$$

Secondly, rather than applying topic models over “bag-of-words” documents, we can first do a part-of-speech (POS) tagging and then apply topic models over “bag-of-POSS” documents. We apply topic models over the source side and target side to derive static features or dynamic features for each rule.

7. Conclusion

In this paper, we propose a simple method to alleviate the over-generation problem of SCFG rules in hierarchical phrase-based translation. By applying topic modeling over the document that consists of all the phrases that X represents, we can learn distributions for each non-terminal X in SCFG rules. In this way, we associate some semantic roles with X in an SCFG rule. By further deriving static features as semantic constraints, we observe that using static features computed from topic models can, to some extent, alleviate the over-generation problem, and improve the performance of hierarchical phrase-based translation.

8. Acknowledgements

This work is partially supported by the DARPA TRANSTAC program under the contract number NBCH2030007. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

9. References

- [1] P. Koehn, F. J. Och, and D. Marcu, “Statistical phrase-based translation,” in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, ser. NAACL ’03, 2003, pp. 48–54.

³<http://homepages.inf.ed.ac.uk/s0562315/progs/>

Table 3: Five example topics from Chinese corpora, each showing top 30 words with the corresponding probabilities in the topic.

Topic 1		Topic 2		Topic 3		Topic 4		Topic 5	
one	0.106429	to	0.434245	your	0.525790	flight	0.064849	feel	0.106028
two	0.096452	from	0.148296	blood	0.034338	time	0.051808	little	0.058829
three	0.084331	boston	0.034482	head	0.024679	first	0.036022	feeling	0.032622
five	0.067770	san_francisco	0.023407	chest	0.021400	flights	0.031301	started	0.019469
four	0.052887	denver	0.023170	leg	0.017910	before	0.026990	drink	0.019065
six	0.045995	atlanta	0.018960	arm	0.017475	leave	0.024210	water	0.017633
eight	0.038834	pittsburgh	0.018062	pressure	0.016183	after	0.023127	coffee	0.016981
seven	0.037152	philadelphia	0.014273	neck	0.015563	morning	0.022940	real	0.014542
nine	0.031375	dallas	0.013793	heart	0.015085	last	0.022746	bit	0.014095
ten	0.026144	fly	0.013493	his	0.011260	around	0.018680	cold	0.013276
zero	0.024377	new_york	0.012911	fingers	0.009957	which	0.018587	hot	0.012439
dollars	0.021626	baltimore	0.011218	hands	0.009791	p_m	0.017527	tea	0.011358
thirty	0.018725	airport	0.008694	eyes	0.009226	day	0.014979	kind	0.010980
ago	0.017001	go	0.008458	legs	0.007857	on	0.014579	felt	0.010875
hundred	0.016926	leaving	0.008227	pulse	0.007461	trip	0.013586	dizzy	0.010427
twenty	0.016623	chicago	0.007574	body	0.007040	next	0.013389	wine	0.010243
days	0.015052	washington	0.007241	lungs	0.006522	a_m	0.011863	beer	0.010030
fifty	0.013997	seattle	0.006882	arms	0.006214	night	0.011760	tired	0.009870
about	0.012965	oakland	0.006696	stomach	0.006008	later	0.011328	steak	0.007496
forty	0.009260	los_angeles	0.006212	nose	0.005956	stop	0.010875	fish	0.007318
years	0.008711	san_diego	0.005803	clothes	0.005719	train	0.010818	ice	0.007258
week	0.008414	toronto	0.005688	vital	0.005394	tomorrow	0.010157	cup	0.007218
thousand	0.007849	washington_d_c	0.005625	abdomen	0.005369	every	0.009943	salad	0.007211
dollar	0.007412	miami	0.004360	toes	0.005212	available	0.009742	with	0.007143
twelve	0.007223	phoenix	0.004103	skin	0.005007	second	0.009729	cream	0.006589
hours	0.007092	kansas_city	0.004045	under	0.004966	airlines	0.009420	fresh	0.006514
fifteen	0.006438	houston	0.003812	ears	0.004965	arrive	0.009268	better	0.006212
number	0.006204	orlando	0.003569	feet	0.004818	round	0.009111	really	0.005809
hour	0.006155	milwaukee	0.003370	shoulder	0.004771	departure	0.008922	juice	0.005679
ninety	0.005936	las_vegas	0.003129	mouth	0.004762	afternoon	0.008597	chicken	0.005563

- [2] D. Chiang, "Hierarchical phrase-based translation," *Computational Linguistics*, vol. 33, no. 2, pp. 201–228, 2007.
- [3] K. Yamada and K. Knight, "A syntax-based statistical translation model," in *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ser. ACL '01, 2001, pp. 523–530.
- [4] M. Galley, M. Hopkins, K. Knight, and D. Marcu, "What's in a translation rule?" in *HLT-NAACL*, 2004, pp. 273–280.
- [5] Y. Liu, Q. Liu, and S. Lin, "Tree-to-string alignment template for statistical machine translation," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ser. ACL-44, 2006, pp. 609–616.
- [6] A. Zollmann and A. Venugopal, "Syntax augmented machine translation via chart parsing," in *NAACL 2006 - Workshop on statistical machine translation*, 2006.
- [7] Z. Huang, M. Cmejrek, and B. Zhou, "Soft syntactic constraints for hierarchical phrase-based translation using latent syntactic distributions," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, Cambridge, MA, October 2010, pp. 138–147. [Online]. Available: <http://www.aclweb.org/anthology/D10-1014>
- [8] J. Pitman and M. Yor, "The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator," *The Annals of Probability*, vol. 25, no. 2, pp. 855–900, 1997.
- [9] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, 2003.
- [10] J. K. Pritchard, M. Stephens, and P. Donnelly, "Inference of population structure using multilocus genotype data," *Genetics*, vol. 155, pp. 945–959, 2000.
- [11] T. L. Griffiths and M. Steyvers, "Finding scientific topics," in *Proceedings of the National Academy of Sciences*, 101 (suppl. 1), 2004, pp. 5228–5235.
- [12] Y. Teh, M. Jordan, M. Beal, and D. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.
- [13] T. S. Ferguson, "A Bayesian analysis of some nonparametric problems," *Annals of Statistics*, vol. 1, no. 2, pp. 209–230, 1973.
- [14] F. Och and H. Ney, "The alignment template approach to statistical machine translation," *Computational Linguistics*, vol. 30, no. 4, pp. 417–449, 2004.
- [15] B. Zhou, B. Xiang, X. Zhu, and Y. Gao, "Prior derivation models for formally syntax-based translation using linguistically syntactic parsing and tree kernels," in *Second Workshop on Syntax and Structure in Statistical Translation (SSST-2)*, 2008.
- [16] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359–393, 1999.
- [17] F. Och, "Minimum error rate training in statistical machine translation," in *Proc. of the 41st Annual Meeting on Association for Computational Linguistics*, 2003, pp. 160–167.
- [18] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2002, pp. 311–318.
- [19] S. Huang and S. Renals, "Modeling topic and role information in meetings using the hierarchical Dirichlet process," in *Machine Learning for Multimodal Interaction V*, ser. Lecture Notes in Computer Science, A. Popescu-Belis and R. Stiefelwagen, Eds. Springer, 2008, vol. 5237, pp. 214–225.