Word Alignment Model with Longer Contexts

Hitoshi Otsuki Masahiro Araki Taro Watanabe
Kyoto Institute of Technology NICT
otsuki@ii.is.kit.ac.jp,araki@kit.jp,taro.watanabe@nict.go.jp

1 Introduction

Word alignment is a crucial component in MT and its quality directly affects the end-to-end MT performance in most cases. The generative word alignment model, i.e. IBM Model[1], is learned in an unsupervised fashion, and considers limited history, i.e. a pair of source and target word. Previous studies incorporated longer histories but little gains were observed mainly due to sparsity issues[5]. To address this issue, we introduce two models; n-gram based context model and skip-gram based context model, both of them take into account longer contexts for disambiguating translations. The sparsity issue incurred by richer contexts are resolved by Kneser Ney(KN) smoothing on fractional counts[10], one of the state-of-the-art performing smoothing techniques. Experimental results on French-English and Chinese-English pairs for word alignment and translations tasks were, to our surprise, negative in any cases.

2 Related Work

2.1 IBM Model

IBM Model is an instance of noisy channel model in which the task of translation is modeled as a generative process[1]. Given a source string $\mathbf{f} = f_1 \cdots f_j \cdots f_J$ and a target string $\mathbf{e} = e_1 \cdots e_i \cdots e_I$, we want to find $\hat{\mathbf{e}}$ which maximizes $Pr(\mathbf{e}|\mathbf{f})$. Here, by using Bayes' theorem, the model is split into two terms,

$$\hat{\mathbf{e}} = \operatorname*{argmax}_{\mathbf{e}} Pr(\mathbf{e}|\mathbf{f}) = \operatorname*{argmax}_{\mathbf{e}} Pr(\mathbf{e}) Pr(\mathbf{f}|\mathbf{e}) \quad (1)$$

The former is called language model, and the latter is called translation model. Now we introduce a hidden variable $\mathbf{a}$ which is a representation of word alignment:

$$Pr(\mathbf{f}|\mathbf{e}) = \sum_{\mathbf{a}} Pr(\mathbf{f}, \mathbf{a}|\mathbf{e}) \quad (2)$$

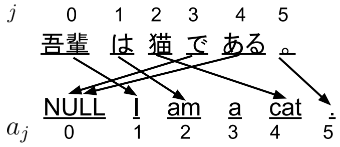

Figure 1: word alignment; a_j is the index of a target word a source word at index j is aligned to .

An example of word alignment is presented in Figure 1. The maximum likely word alignment given a pair of sentences $\mathbf{f}$ and $\mathbf{e}$ is computed by replacing the summation in equation 2 with maximization which is called Viterbi alignment. IBM Model has defined several models by varying the richness and complexity in representation, and we will concentrate on two simpler models, Model 1 and 2 in particular. $Pr(\mathbf{f}, \mathbf{a}|\mathbf{e})$ in Equation 2 is divided into several components as follows:

$$Pr(\mathbf{f}, \mathbf{a}|\mathbf{e}) \\ = Pr(J|\mathbf{e}) \prod_{j=1}^{J} Pr(a_j|a_1^{j-1}, f_1^{j-1}, J, \mathbf{e}) Pr(f_j|a_1^{j}, f_1^{j-1}, J, \mathbf{e}) \quad (3)$$

It is assumed that the length model $Pr(J|\mathbf{e})$ is independent of $\mathbf{e}$ and that $Pr(f_j|a_1^j, f_1^{j-1}, J, \mathbf{e})$, called lexicon model, depends only on f_j and e_{a_j} which we denote as $t(f_j|e_{a_j})$. Under Model 1, the alignment model $Pr(a_j|a_1^{j-1}, f_1^{j-1}, J, \mathbf{e})$ is assumed to be uniform $\frac{1}{I+1}$ whereas the model is dependent on just a few variables under Model 2 via $a(a_j|i, I, J)$. In this work, we employ a variant of Model 2 which strongly prefers alignments to be close to the diagonal[4]. As an extension of IBM model 1, triplet lexicon model consider two words in the target side, e_i and e_k, in the lexicon model as a way to disambiguate translations by exploring additional contexts[5].
Training of these models are performed iteratively using EM algorithm, in which posterior probability $P(\mathbf{a}|\mathbf{f}, \mathbf{e}; \theta^{(t)})$ for each sentence pair is computed using the current parameters $\theta^{(t)}$, and optimal parameters $\theta^{(t+1)}$ is obtained by maximizing the expectation of joint probabilities of $\mathbf{f}$ and $\mathbf{a}$ given $\mathbf{e}$ over $P(\mathbf{a}|\mathbf{f}, \mathbf{e}; \theta^{(t)})$[1],

$$\theta^{(t+1)} = \operatorname*{argmax}_{\theta} \sum_{<\mathbf{f},\mathbf{e}>} \sum_{\mathbf{a}} P(\mathbf{a}|\mathbf{f}, \mathbf{e}; \theta^{(t)}) \log P(\mathbf{f}, \mathbf{a}|\mathbf{e}; \theta) \quad (4)$$

2.2 Smoothing

Smoothing is a popular technique in machine learning and used for estimating parameters even for unseen event and for avoiding overfitting to a training data. One of the popular choice of smoothing is regularization which directly prevents from overfitting toward training data, for instance by using ℓ_0 prior[9].
Another form of smoothing is backoff smoothing primarily used in n-gram language models in which counts are discounted and the subtracted counts are reserved for unseen data. More formally, given an n-gram $\mathbf{u}w$ with its frequency $c(\mathbf{u}w)$ and a certain discounted amount $D(\mathbf{u}w)$, the probability of observing w given contexts $\mathbf{u}$ is represented as follows:

$$p(w|\mathbf{u}) = \frac{c(\mathbf{u}w) - D(\mathbf{u}w)}{c(\mathbf{u}\,\bullet\,)} + \alpha(\mathbf{u})p(w|\mathbf{u}') \qquad (5)$$

where $\alpha(\mathbf{u})$ is used to ensure the probability sum to 1. In absolute discounting and KN smoothing[2] $D(\mathbf{u}w)$ is a constant. The probability $p(w|\mathbf{u}')$ is a lower order probability which truncate $\mathbf{u}$ to $\mathbf{u}'$. Under KN smoothing, the probability is taken to satisfy marginal constraint, leading to:

$$p(w|\mathbf{u}') = \frac{n_{1+}(\,\bullet\,\mathbf{u}'w)}{n_{1+}(\,\bullet\,\mathbf{u}'\,\bullet\,)} \qquad (6)$$

Recursive smoothing further smooths $n_{1+}(\,\bullet\,\mathbf{u}'w)$ to lower order. Uniform distribution $\frac{1}{|V|}$ is taken to end the recursion, where $|V|$ denotes the size of vocabulary. Modified KN, which is reported to be one of the best performing smoothing technique[2], use three different discounts D_1,D_2 and D_{3+}, that are applied to n-gram with one, two, three or more counts.

The original (Modified) KN smoothing is only applicable to integer counts. In EM algorithm, however, collected counts are fractional, which makes KN smoothing inapplicable. KN smoothing on expected counts[10] extended KN smoothing to fractional counts by incorporating count distribution $p(c(\mathbf{u}w) = r)$ which is the pobability that n-gram $\mathbf{u}w$ appears r times.

3 Alignment Model with Long Contexts

One of the drawback of IBM Model is the limited context represented in lexicon model. As a way to disambiguate correspondences between source and target words, we present two models to consider longer contexts in the target side, namely n-gram based model and skip-gram based model. However, longer contexts directly represented in lexicon models incur severe sparsity problems. Thus, we introduce modified KN smoothing for fractional count as a way to backoff to shorter contexts during the estimation of parameters in M-step.

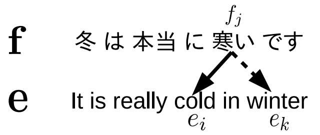

Figure 2: triplet lexicon model: source word f_j is aligned to e_i, e_k

3.1 n-gram based model

n-gram based model extends the lexicon model by considering previous $n-1$ consecutive words in the target side as follows:

$$\begin{aligned} p(\mathbf{f}|\mathbf{e}) =& p_t(f_j|\mathrm{NULL}, \ldots, \mathrm{NULL})p_0 + \\ & (1-p_0)\prod_{j=1}^{J}\sum_{i=1}^{I} p_t(f|e_{i-n}\cdots, e_{i-1}, e_i)\cdot a(i|j, I, J) \end{aligned} \qquad (7)$$

where p_0 is null alignment probability and $a(i|j, I, J)$ is an alignment model which favors diagonal alignment points. Note that the n-gram based model is a very simple extension of IBM Model and exactly the same EM algorithm can be used to estimate the parameters despites the growth of parameters of the lexicon model.

3.2 skip-gram based model

Skip-gram based model is triplet lexicon model[5] combined with alignment model which favors diagonal alignment points. In this model, each element in word alignment $\mathbf{a}$ is a pair of indices in target sentence, $a_j = (i, k)$, where $i < k$ with the first element used as a primary index for alignment link and the second element used as secondary index to represent context, as shown in Figure 2. Like the original paper of triplet lexicon model, we define $p_{all}(f_j, e_1^I)$, the probability of a source word f_j given the whole target sentence e_1^I as follows[1]:

$$\begin{aligned} & p_{all}(f_j|e_1^I) = p_t(f_j|\mathrm{NULL}, \mathrm{NULL})\cdot p_0 + \\ & (1-p_0)\sum_{i=1}^{I}\sum_{k=i+1}^{I+1}\frac{1}{I+1-i}p_t(f_j|e_i, e_k)\cdot a(i|j, I, J) \end{aligned} \qquad (8)$$

where p_0 is null alignment probability. In skip-gram based model, Viterbi alignment is obtained as fol-

[1] e_{I+1} denotes an end-of-sentence character

lows:

$$a_j = \operatorname*{argmax}_{0\le i\le I} p_t(f_j|\mathrm{NULL},\mathrm{NULL}) \cdot p_0 \cdot \delta(i=0) +$$
$$(1-p_0)\sum_{k=i+1}^{I+1} \frac{1}{I+1-i} p_t(f_j|e_i,e_k)\cdot a(i|j,I,J)\cdot\delta(i\neq 0) \tag{9}$$

where $\delta(\cdot)$ evaluates to 1 if the condition $\cdot$ holds; otherwise evaluates to 0.
EM algorithm is similar to that of IBM Model, except that we need to check secondary index k which is larger than primary index i. Sum over k is taken to properly obtain Viterbi alignment.

4 Experiments

4.1 Setting

We carried out experiments on two tasks for two language pairs: French-English and Chinese-English. First task is to measure the quality of word alignment, for which we used Hansards[2], French-English corpus, consisting of 1.1M parallel sentences pairs as training data and 447 word-aligned sentence pairs for evaluation. Second task is to measure the translation quality. For this, we used Europarl+news commentary, another French-English corpus, from wmt12 translation task[3], consisting of 2M+137K of parallel corpus as training data, 3003 sentence as development data and 2489 sentences for evaluation. Also we used FBIS corpus, Chinese-English corpus, which includes 138K parallel sentences as training data, 878 sentences for development data and 919 sentences for test data with 4 references each.

The aligners are modified version of fast_align[4] of cdec[4] with giza-kn[5] for the code base for KN smoothing on fractional counts[10]. For all tasks, we trained both n-gram based aligner, n ranging from 1 to 3 or 4, and skip-gram based aligner. For each corpus, we conducted experiments on different corpus sizes. For Hansards, a 1/10, 1/5, 1/2 and all of corpus were used for the experiments. For FBIS, 1/5, 1/2 and all of the corpus were used for the experiment. And for Europarl+news commentary, a 1/10, 1/5, 1/2 of the corpus were used. Due to memory and time constraint, 5 iterations was used for training with Hansards and Europarl+news commentary and 10 iterations for FBIS, and window size was set to 5 in skip-gram based aligner.

We aligned on both directions of a language pair and then symmetrized them using grow-diag-final-and method. Word alignment qualities for Hansards were measured by AER[7]. For the other data set, we measured translation qualities by BLEU[8] using a phrase-based MT of Moses[6] tuned by batch-MIRA[3].

4.2 Results

Our experimental results showed that adding context degraded AER and BLEU scores. Experimental results on word alignment qualities measured by AER are summarized in Table 1. For n-gram based aligner, AER basically got worse as the number of contexts increases in both KN and maximum likelihood estimate(ML). While almost no differences were observed for 2-gram and above, AER drops significantly when a context is added to the 1-gram model. The drop is even more significant in ML than KN, demonstrating up to 15% degradation. Skip-gram base aligner scored better than 2-gram aligner, 5% better in KN and 8% in ML at least. We also observe that Kneser ney smoothing surpress the worsening of the results compared to ML. The results for translation qualities are presented in Table 2. We can see that skip-gram aligner performed poorer than 2-gram aligner in FBIS, up to 0.69 difference in BLEU score when KN is used, and up to 0.65 difference when ML is used. On the other hand, skip-gram based model outperformed 2-gram model in Europarl + news commentary, though it falls behind 1-gram model. Like AER in Table 1, BLEU generally drops as the number of context increases.

4.3 Discussion

We observed that additional contexts had negative impact for the AER and BLEU score. Figure 3 shows an example of word alignment results from 1-gram aligner and 2-gram aligner under Hansard. In this example, “threat” is aligned to “menace”, which is the correct alignment, when 1-gram aligner is used, and is aligned to “pour” when 1-gram aligner is used. We extracted the translation table produced after training 2-gram aligner with Hansard corpus in Table 3. Table 3 shows that the probability that “menace pour” is linked to “threat” is very high, even though we want the lexical translation model to link “menace pour” to “to” or “for” in almost all cases, not “threat”. We confirmed that “pour” is most likely be translated to “to” or “for” in 1-gram aligner. This implies that adding context to lexical translation model incurs more parameters to be estimated to the extent KN smoothing can't handle. Therefore, if a n-gram appears only a few times in a training data, relatively low probability is assigned to the parameter even if a word consisting the n-gram appears frequently, as in “serieuse menace” → “threat”. Indeed, “menace pour” appeared

[2] http://www.isi.edu/natural-language/download/hansard/

[3] http://www.statmt.org/wmt12/translation-task.html

[4] https://github.com/redpony/cdec/word-aligner

[5] https://github.com/hznlp/giza-kn

Table 1: AER of n-gram aligner and skip-gram aligner, trained on Hansards corpus with 5 EM iterations

	Kneser Ney				Maximum Likelihood			
model	1/10	1/5	1/2	all	1/10	1/5	1/2	all
1-gram	**15.09**	**14.56**	**14.06**	**13.90**	**15.76**	**14.92**	**14.20**	**14.06**
2-gram	22.59	23.28	24.00	24.89	29.73	29.43	29.28	28.77
3-gram	23.43	24.16	25.09	26.29	32.12	32.85	32.96	32.79
4-gram	23.30	23.95	24.76	26.08	33.43	33.03	32.16	32.74
skip	17.22	17.34	17.54	17.52	21.70	21.26	20.44	20.01

Table 2: BLEU of n-gram aligner and skip-gram aligner, trained on FBIS corpus with 10 EM iterations and Europarl + news commentary with 5 EM iterations

	FBIS						Europarl + news commentary					
	Kneser Ney			Maximum Likelihood			Kneser Ney			Maximum Likelihood		
model	1/5	1/2	all	1/5	1/2	all	1/10	1/5	1/2	1/10	1/5	1/2
1-gram	**22.1**	**26.05**	**28.11**	**20.80**	**24.46**	**26.89**	**23.01**	**23.69**	**24.17**	**22.87**	**23.58**	**24.07**
2-gram	19.77	23.62	26.22	14.69	18.34	20.7	22.42	23.00	23.55	20.58	21.28	22.14
3-gram	17.87	21.63	24.37	13.85	17.39	19.63	22.14	22.69	23.21	19.63	20.56	21.46
4-gram	17.94	20.94	23.16	13.5	17.30	19.28						
skip	19.33	22.93	25.99	14.38	17.69	20.56	22.69	23.52	23.86	21.09	22.17	23.12

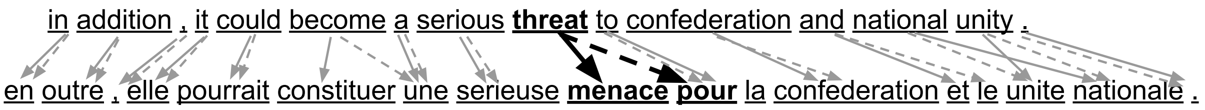
in addition , it could become a serious threat to confederation and national unity .
en outre , elle pourrait constituer une serieuse menace pour la confederation et le unite nationale .

Figure 3: Alignment points between a English sentence and a French sentence; dashed line is the alignments from 2-gram aligner, and normal line is the alignments by 1-gram aligner

Table 3: translation table of hansard corpus, learned by 2-gram aligner

f	e_{i-1}, e_i	$t(f \mid e_{i-1}, e_i)$
to	menace pour	0.476080
threat	menace pour	0.298240
threat	serieuse menace	0.121866

more than 100 times in the training corpus, while “serieuce menace” appeared only 3 times.

5 Conclusion

We introduced two translation model, n-gram based model and skip-gram based model. n-gram based model considers previous context in the lexicon translation model where as skip-gram based model considers words not only from local context but also from global context. Our experiments showed that adding contexts has negative impact on both of the alignment quality and translation quality, though previous studies insisted on positive results under more simpler triplet lexicon model. The extra contexts might cause huge data sparsity issue than we have expected which cannot be handled by simply introducing a state-of-the-art backoff smoothing of KN on expected counts. In the future, we would like to investigate the issue using more sophisticated method, such as non-parametric Bayesian to properly handle longer contexts.

References

[1] Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311, 1993.

[2] Stanley F Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics, 1996.

[3] Colin Cherry and George Foster. Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 427–436. Association for Computational Linguistics, 2012.

[4] Chris Dyer, Victor Chahuneau, and Noah A Smith. A simple, fast, and effective reparameterization of ibm model 2. In *HLT-NAACL*, pages 644–648. Citeseer, 2013.

[5] Saša Hasan, Juri Ganitkevitch, Hermann Ney, and Jesús Andrés-Ferrer. Triplet lexicon models for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 372–381. Association for Computational Linguistics, 2008.

[6] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL ’07, pages 177–180, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

[7] Franz Josef Och and Hermann Ney. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 440–447. Association for Computational Linguistics, 2000.

[8] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[9] Ashish Vaswani, Liang Huang, and David Chiang. Smaller alignment models for better translations: unsupervised word alignment with the l 0-norm. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 311–319. Association for Computational Linguistics, 2012.

[10] Hui Zhang and David Chiang. Kneser-ney smoothing on expected counts.