International Journal of Advanced Intelligence
Volume 4, Number 1, pp.87-94, December, 2012.

AIA
International
Advanced
Information
Institute

A Fully Bayesian Inference for Word Alignment

Li Zezhong*
Department of Computer Science, Ritsumeikan University Shiga, Japan
lizezhonglaile@163.com

Hideto Ikeda
Department of Computer Science, Ritsumeikan University Shiga, Japan
hikeda@ritsumei.ac.jp

Received (11, July, 2012)
Revised (9, December, 2012)

We present an approximative IBM Model 4 for word alignment. Different with the most widely-used word aligner GIZA++, which implements all the 5 IBM models and HMM model in the framework of Expectation Maximum (EM), we adopt a full Bayesian inference which integrates over all possible parameter values, rather than estimating a single parameter value. Empirical results show promising improvements in alignment quality as well as in BLEU score for the translation performance over baselines.

Keywords: Bayesian inference; Word alignment; Statistical machine translation.

1. Introduction

Word alignment can be defined as a procedure for detecting the corresponding words in a bilingual sentence pair. One of the notorious criticisms of word alignment is the inconsistence between the word alignment model to the phrase based translation model. In this paper, we have no intention to avoid mentioning this inherent weakness of word alignment, but we would say, as far as we know, word alignment is a fundamental component for most of the SMT systems. Phrase or the other higher level translation knowledge is extracted based on the word alignment, which is called two-stage approach. And even for approaches of so-called direct phrase alignment, they can rarely abandon word alignment thoroughly. Because of the computation complexity of phrase alignment, word alignment is usually used to constrain the inference.[1] DeNero proposes a relative pure joint phrase model but still uses the word alignment as initialization and smoothing, which shows the least dependency on word alignment.[2] Neubig uses Bayesian methods and Inversion Transduction Grammar for joint phrase alignment,[3] and the base distribution for the Dirichlet Process [5] prior is constructed by the word alignment model. Therefore, word alignment is well worth concern. Our hope is to induce a better word alignment by

*1-1-1, Noji-Higashi, Kusatsu, Shiga, Japan.

utilizing the state-of-the-art learning technology, and establish a better baseline for the word level alignment models.

Bayesian inference, the approach we adopt in this paper, has been broadly applied to various learning of latent structure. Goldwater points out that two theoretical factors contribute to the superiority of Bayesian inference.[7] First, integrating over parameter values leads to greater robustness in decision. One of the problems that trouble EM algorithm is over-fitting. Moore discusses details of how a "Garbage collector" is generated.[11] He also suggests a number of heuristic solutions, but Bayesian inference can offer a more principled solution. The second factor is that the integration permits the use of priors favoring sparse distributions, which proved to be more consistent with nature of natural language. Another practical advantage is that the implementation can be much easier than EM.[12]

In the following sections, we will have a review for IBM Model 4 in Section 2, and reformulate it into a simpler and Bayesian form in Section 3. Section 4 gives the Bayesian inference, and Section 5 reports results of experiment. Section 6 compares related research, and Section 7 concludes.

2. IBM Model 4

Model 4 is a fertility-based alignment model, and can be viewed as the outstanding representative of all the IBM translation models. The model can be expressed as

$$\begin{aligned} P(F, A|E; n, t, d) &= P_\phi(\phi_0^I|E; n) P_\tau(\tau_0^I|\phi_0^I, E; t) P_\pi(\pi_0^I|\phi_0^I, \tau_0^I, E; d) \\ &= n_0(\phi_0|\sum_{i=1}^{I}\phi_i)\prod_{i=1}^{I} n(\phi_i|e_i)\prod_{i=0}^{I}\prod_{k=1}^{\phi_i} t(\tau_{ik}|e_i)\frac{1}{\phi_0!}\prod_{i=1}^{I}\prod_{k=1}^{\phi_i} p_{ik}(\pi_{ik}) \end{aligned} \tag{1}$$

where

$$n_0(\phi_0|\sum_{i=1}^{I}\phi_i) = \binom{\sum_{i=1}^{I}\phi_i}{\phi_0} {p_0}^{\sum_{i=1}^{I}\phi_i - 2\phi_0} {p_1}^{\phi_0} \tag{2}$$

$$p_{ik}(\pi_{ik}) = \begin{cases} d_1(j - c_{\rho_i}|\mathfrak{A}(e_{\rho_i}), \mathfrak{B}(\tau_{i1})) & if\ k = 1 \\ d_{>1}(j - \pi_{ik-1}|\mathfrak{B}(\tau_{ik})) & if\ k > 1 \end{cases} \tag{3}$$

P_ϕ, P_τ and P_π denote fertility model, lexical model and distortion model respectively, and their parameters can be described as n, t, and d. More details can be found in Brown et al.[4]

3. Bayesian Model

Our Bayesian model almost repeats the same generative scenarios shown in the previous section, but puts an appropriate prior for the parameters in the model. That is, parameter will be treated as variable, which makes a significant difference to the traditional MLE or MAP approaches. In our proposed Bayesian setting, the fertility

ϕ and translation f for each target word e, both of which follow a Multinominal distribution, will be treated as a random variable with a prior, and Dirichlet distribution is a natural choice for them, since it is conjugate to the Multinominal distribution. Since we can not specify the dimensions of the above distributions in advance, one solution is to take advantage of the nonparametric prior. Here we use the Dirichlet Process (DP) which can ensure that the resulting distributions concentrate their probability mass on a small number of fertilities or translation candidates while retaining reasonable probability for unseen possibilities.

$$n_e \sim DP(\alpha, Poisson(1, \phi)) \tag{4}$$
$$\phi|e \sim n_e \tag{5}$$
$$t_e \sim DP(\beta, T_0(f|e)) \tag{6}$$
$$f|e \sim t_e \tag{7}$$

In the above distribution formulas, n_e denotes the fertility distribution for e, and hyperparameter α is a concentration parameter which affects the variance of the draws. We make $Poisson(1, \phi)$ as the base distribution for fertility which encodes our prior knowledge about the properties of fertilities. Namely, high fertility should be discouraged except that there is enough evidence. $\lambda(e)$ denotes the expected fertility for e, and for simplicity, we assign 1 as the value of expected fertility for all the words. t_e is a translation distribution for e, and β is the concentration parameter. As for base distribution T_0, shown as:

$$T_0(f|e) = \sum_{et,ft} p(et|e)p(ft|et)p(f|ft) \tag{8}$$

where et denotes e's Part-of-Speech (henceforth POS), and ft denotes f's POS. $p(ft|et)$ is a POS translation model, $p(et|e)$ is a transition probability from word to POS, and $p(f|ft)$ is a uniform distribution (over word types tagged with ft) for each word f. T_0 encodes such a prior knowledge: POS provides clues for the alignment.

While our Bayesian model still has other free parameters, we still use p_0 and p_1 as parameters to model fertility for e_0 as same as in IBM models, but we fix them to reasonable values in order to focus on learning for the other distributions. As for the distortion model, we simply adopt a distance penalty (not including the distortion for words generated by e_0) shown as follows

$$p_\pi(A) \propto \frac{1}{\phi_0!} \prod_{j=1, a_j \neq 0}^{J} b^{|j - prev(j)|} \tag{9}$$

$$prev(j) = \begin{cases} \pi_{\rho_i \phi_{\rho_i}} & if\ k = 1 \\ \pi_{ik-1} & if\ k > 1 \end{cases} \tag{10}$$

where b is a fixed value less than 1, $prev(j)$ means the position of predecessor for f_j. ρ_i denotes the first position to the left of e_i for which has a non-zero fertility, and π_{ik} is the position of word τ_{ik} for permutation π. The first part of our distortion formula models the distortion procedure for words generated by e_0, which uses the same strategy as IBM models that all these words are positioned only after the nonempty positions have been covered. Therefore, there are $\phi_0!$ ways to order the ϕ_0 words.

Due to the above simplification for fertility model, we will see a more convenient inference in following sections. Another theoretical reason is that we do not expect a skewed distribution for the above parameters as same as the fertility and lexical models. Therefore, it is unnecessary to put a prior for these parameters.

4. Bayesian Inference

A frequent strategy to infer the posterior distribution is Gibbs sampling.[12] For our concerned word alignment, instead of sampling the parameters explicitly, we sample the alignment structure directly with the parameters marginalized out. Then the Gibbs sampler is converted into a collapsed Gibbs sampler, and we have

$$P(F, A|E; \alpha, \beta) = \int_{n,t} P(F, A, n, t, d|E; \alpha, \beta) \tag{11}$$

where n comprises all the n_e for each e, and t comprises all the t_e. d does not need integral since we do not treat this parameter as a random variable, and will be replaced by constant b in the left part of the integral formula. Due to the collapsed sampler, we need not sample the parameters explicitly, but directly sample the latent alignment structure in condition of fixed α and β. Our collapsed Gibbs sampler works by sampling each component of vector a alternatively. The probability for a new component value when the other values are fixed can be written

$$\begin{aligned} P(a_j|a_{\neg j}, F, E;\ \alpha, \beta) \propto & \\ & P_\phi(a_j|a_{\neg j}, F, E; \alpha, \beta) P_\tau(a_j|a_{\neg j}, F, E; \alpha, \beta) P_\pi(a_j|a_{\neg j}, F, E; \alpha, \beta) \end{aligned} \tag{12}$$

where $a_{\neg j}$ denotes the alignment exclude a_j. P_ϕ, P_τ and P_π represent fertility, translation and distortion sub-models respectively. The probability of new sample can be calculated according to the three sub-models. This calculation is very similar with the procedure that finds the neighbour alignments in the E-step of EM, but in a way metaphorized as Chinese Restaurant Process instead of using fixed parameters.[12] First, we will investigate the translation model. Thanks to the exchangeability, we can write

$$P_\tau(a_j|a_{\neg j}, F, E; \alpha, \beta) \propto \frac{Count(e_{a_j}, f_j) + \beta T_0(f_j|e_{a_j})}{\Sigma_f Count(e_{a_j}, f) + \beta} \tag{13}$$

where $Count(e, f)$ is the number of links between word pair (e, f) in the other part of this sentence pair and other sentence pairs in the training corpus.

As for the fertility model, because of the special treatment of the fertility for e_0, two cases should be considered. In the first case $a_j! = 0$,

$$P_\phi(a_j|a_{\neg j}, F, E; \alpha, \beta) \propto \frac{Count(e_{a_j}, \phi_{a_j} + 1) + \alpha Poisson(1, \phi_{a_j} + 1)}{Count(e_{a_j}, \phi_{a_j}) + \alpha Poisson(1, \phi_{a_j})} \tag{14}$$

where $Count(e, \phi)$ is the frequency of cases where word e has a fertility ϕ, and the denominator encodes the fact that the new assignment will cause an instance of word-fertility to be removed from the cache as the new word-fertility is added. And in the second case, $a_j = 0$. As is described in the previous section, the fertility for empty word is not decided by itself, but decided by the number of words generated by nonempty words, which follows a binominal distribution. So we can infer

$$P_\phi(a_j = 0|a_{\neg j}, F, E; \alpha, \beta) \propto \frac{n_0(\phi_0 + 1|\sum_{i=1}^{I} \phi_i)}{n_0(\phi_0|\sum_{i=1}^{I} \phi_i)} = \frac{(\sum_{i=1}^{I} \phi_i - \phi_0)p_1}{(\phi_0 + 1)p_0} \tag{15}$$

The calculation for the distortion model is more direct since it is unnecessary to consider the cache model. Because of the special treatment for distortion of words aligned with the empty word, we also need to take account two cases, as for the first case, $a_j! = 0$

$$P_\pi(a_j|a_{\neg j}, F, E; \alpha, \beta) \propto b^{|j - prev(j)| + |next(j) - j| - |next(j) - prev(j)|} \tag{16}$$

where the exponent means 3 distortions are changed, and $next(j)$ is subject to $j == prev(next(j))$. In the second case, where $a_j = 0$, we just need to consider the probability of a permutation of ϕ_0 words in the remained uncovered positions. Notice that, the fertility value changes from ϕ_0 to $\phi_0 + 1$ after this new assignment, then we have

$$P_\pi(a_j = 0|a_{\neg j}, F, E; \alpha, \beta) \propto \frac{\phi_0!}{(\phi_0 + 1)!} = \frac{1}{\phi_0 + 1} \tag{17}$$

The final probability for the new derivation should combine all the above influence factors, and the production of all the three factors as the final probability. The algorithm is described in Table 1. To accelerate the convergence, we use HMM based Viterbi alignment as an initialization. After burn-in iterations, we begin to collect alignment counts from the samples.

5. Experiments

All the corpus we used is Chinese-English corpus in patent domain, which is released by NTCIR9.[15] We select 350K sentence pairs as training corpus, and 1000 pairs as

Table 1. Gibbs sampling for word alignment.

```
For each sentence pair (E, F) in corpus
  Initialize alignment
For each generation
  For each sentence pair (E, F) in corpus
    For each j in [1, |F|]
      For each i in [0, |E|]
        calculate p(a_j = i|a_¬j, F, E; α, β)
      Normalize p(a_j|a_¬j, F, E; α, β)
      Sample a new value for a_j; update the cache count
    If (Current generation ≥ Burn-in)
      Save alignment for (E, F)
```

development set. We also annotate 300 word aligned sentence pairs to evaluate the quality of word alignment, and select 2000 bilingual pairs as the test set for translation. Before running our Bayesian aligner, we should estimate the parameters in T_0. We tagged the training corpus using some POS taggers, and replace each word by its POS to get a POS parallel corpus. Then, we ran IBM model 1 on the POS corpus to get the POS translation probabilities. Through dividing the number of occurrences of the word-tag pair (e, et) by the number of occurrences of e, we can get $p(et|e)$. Suppose word f is tagged with ft at least once in the training corpus, then $p(f|ft)$ is equal to the result of dividing 1 by the number of unique words tagged with ft; otherwise, $p(f|ft)$ is 0.

To contrast our approach with GIZA++, we need the Viterbi alignment extracted from the multiple samples, and one strategy is to assign each a_j as the most frequent value in the collected samples. We set 1000 as the number of total iterations and 0 as the burn-in value, and configure α and β with varying values. We run GIZA++ in the standard configuration (Training scheme is abbreviated as $1^5H^53^34^3$). Both of the above two approaches need run in two directions and symmetrization. Table 2 shows the comparison of AER between GIZA++ (EM) and our Bayesian model. When $\alpha = 1$ and $\beta = 100$, our proposed approach can get the best performance, which reveals a satisfying improvement for alignment quality in terms of AER, with a reduction of 3.41% over GIZA++.

For translation experiments, we use Moses as our decoder,[10] and use SRILM to train 4-grams language models on both sides of the bilingual corpus. As is shown in Table 3, we can see that the Bayesian approach outperforms EM approach in both directions, which proves the effectiveness of our proposed approach.

6. Related Work

Our approach is similar with Coskun in spirit to Bayesian inference,[9] where it places a prior for the model parameters and adopts a collapsed sampler, but they take Model 1 as the inference object, which we suppose somewhat harsh. Zhao proposes a brief fertility based HMM model,[8] which also decreases the complexity of Model

Table 2. Performance of Word Alignment.

Method	AER
EM(GIZA++)	16.12%
Bayesian($\alpha = 0.5$, $\beta = 100$)	13.43%
Bayesian($\alpha = 1$, $\beta = 100$)	**12.71%**
Bayesian($\alpha = 1.5$, $\beta = 100$)	13.74%
Bayesian($\alpha = 1$, $\beta = 50$)	15.04%
Bayesian($\alpha = 1$, $\beta = 200$)	12.98%

Table 3. Performance of Final Translation (BLEU-4).

Method	Chinese-English	English-Chinese
EM(GIZA++)	0.2766	0.2964
Bayesian($\alpha = 0.5$, $\beta = 100$)	0.2787	0.2993
Bayesian($\alpha = 1$, $\beta = 100$)	**0.2798**	**0.3011**
Bayesian($\alpha = 1.5$, $\beta = 100$)	0.2781	0.2986
Bayesian($\alpha = 1$, $\beta = 50$)	0.2778	0.2978
Bayesian($\alpha = 1$, $\beta = 200$)	0.2795	0.3003

4 but keeps the fertility as a component of modeling. But they do not place any prior on the parameters, which can be viewed as a stochastic EM. They also assume fertility follows a Poisson distribution, while ours adopts a DP prior and Poisson distribution as the base distribution in the DP prior. Darcey et al. use variational Bayes which closely resembles the normal form of EM algorithm to improve the performance of GIZA++, as well as the BLEU score.[14]

7. Conclusions and Future Work

We have described an approximative IBM model 4 for word alignment, and adopt Bayesian inference which currently is a promising replacement for EM and already broadly applied for various tasks in the field of NLP. Our pilot experiment shows a higher AER for word alignment as well as a modest improved BLEU score for translation. Our current research focuses on phrase extraction and reordering from multiple alignment samples generated by our Bayesian inference, and we expect a better performance.

References

1. Hao Zhang, et al. Bayesian Learning of Non-compositional Phrases with Synchronous Parsing. In *Proceedings of ACL-HLT*, pp. 97-105, 2008.
2. John DeNero, Alexandre Bouchard Cote, Dan Klein. Sampling Alignment Structure Under a Bayesian Translation Model. In *Proceedings of EMNLP*, pp. 314-323, 2008.
3. Graham Neubig, et al. An Unsupervised Model for Joint Phrase Alignment and Extraction. In *Proceedings of ACL*, pp. 632-641, 2011.
4. Peter F. Brown et al. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263-311, 1993.

5. Thomas S Ferguson. A Bayesian Analysis of Some Nonparametric Problems. In *Annals of Statistics*, 1973.
6. Franz Josef Och, Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. In *Computational Linguistics*, 29(1):19-51, 2003.
7. Sharon Goldwater, Tom Griffiths. A Fully Bayesian Approach to Unsupervised Part-of-Speech Tagging. In *Proceedings of the ACL*, pp. 744-751, 2007.
8. Shaojun Zhao, Daniel Gildea. A Fast Fertility Hidden Markov Model for Word Alignment Using MCMC. In *Proceedings of EMNLP*, pp. 596-605, 2010.
9. Coskun Mermer, Murat Saraclar. Bayesian Word Alignment for Statistical Machine Translation. In *Proceedings of ACL*, pp. 182-187, 2011.
10. Philipp Koehn et al. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of ACL*, pp. 177-180, 2007.
11. Robert C. Moore. Improving IBM Word Alignment Model 1. In *Proceedings of ACL*, pp. 518-525, 2004.
12. Philip Resnik, Eric Hardisty. Gibbs Sampling for the Uninitiated. *Technical report*, University of Maryland, 2010.
13. Daniel Marcu and Daniel Wong. A Phrase-based, Joint Probability Model for Statistical Machine Translation. In *Proceedings of EMNLP*, pp. 133-139, 2002.
14. Darcey Riley, Daniel Gildea. Improving the Performance of GIZA++ Using Variational Bayes. University of Rochester. Technical Report, 2010.
15. Tetsuya Sakai, Hideo Joho. Overview of NTCIR-9. In *Workshop of NTCIR-9*, pp. 559-578, 2011.

Li Zezhong (Member)

He is currently a doctor student in Department of Computer Science, Ritsumeikan University, His main research interests include Machine Translation and Natural Language Processing.

Hideto Ikeda (Member)

He received the PhD from Hiroshima University. Dr. Ikeda is currently a professor at Ritsumeikan University. His main research interests include Database, eLearning, Machine Translation and Natural Language Processing.