Homework 2

Junjie Hu
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
junjieh@andrew.cmu.edu

Abstract

In this assignment, we build the statistical phrase-based machine translation system [3] for German-English corpus. We build the alignment model using IBM 1, extract phrase using the alignment and construct the weighted finite state transducer (WFST). We further explore the pseudo-count method in [4] and an initialization trick to improve the word alignment of the IBM Model 1. We further compare both variants of IBM Model1 with the original IBM Model 1 and analyse the experimental results. Our best model achieve 18.08 BLEU score on the test set and 18.64 BLEU score on the validation set.

1 Method

In this section, we detail the implementations of our machine translation system.

1.1 IBM Model 1

IBM Model 1 [2] is a popular word alignment model. We implement the original IBM Model 1 and train the model by EM algorithm. In the E step, we estimate the expected counts given the latent alignment D. In the M step, we apply maximum likelihood estimation to update the model parameter θ_{f_j,e_i} which specifies $P(f_j|e_i;\theta)$ in the current model.

1.2 Phrase Extraction

Since the extracted phrase table is every large, we do not expand the source phrase on the left side and right side, which means we remove the implementation in Lines 11-18 in Algorithm 6. We find that the number of extracted phrases are large enough and we can still obtain a relatively high BLEU score on both validation and test set. We further filter out the rare phrase pairs that appear only once in the training corpus. This dramatically reduce the number of extracted phrase pairs.

1.3 Open FST

We use the open source toolkit (Open FST [1]) to construct our WFST.

2 Modifications

2.1 Smoothing

We apply a pseudo-code method in [4] to obtain a smooth estimate of the probability $P(f_j|e_i)$. That is, the model parameter θ_{f_j,e_i} can be updated by Eq. 1 in each M step.

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

Table 1: BLEU score on German-English corpus

Method	IBM1	IBM1-Smooth	IBM1-Initialization
Dev	18.17	18.64	18.16
Test	17.61	18.08	17.60

$$\theta_{f_j,e_i} = \frac{C(f_j, e_i) + n}{C(e_i) + n \cdot |V|} \tag{1}$$

where $C(f_j, e_i)$ counts the number of times that f_j and e_i appear in the training corpus, $C(e_i)$ counts the number of times that e_i appears in the training corpus, n is the added count for each target word and $|V|$ is the number of distinct words observed in the target language. Since we do not want the vocabulary size of the target language to be too large and we know that the target language can have many rare words that we have never observed in the test/valid corpus, we can simply select a reasonable number for $|V|$. In this report, we empirically set $n = 2$ and $|V| = 2,000$.

2.2 Initialization

Rather than estimating θ_{f_j,e_i} by the uniform distribution over the target vocabulary, we also try to get a good initialization of θ_{f_j,e_i} by frequency estimation in Eq. 2.

$$\theta_{f_j,e_i} = \frac{C(f_j, e_i)}{C(e_i)} \tag{2}$$

3 Experiment

We conduct experiments to translate German to English given the provided dataset. We run the original IBM1 and its variants (i.e., IBM1-Smooth and IBM1-Initialization) and report the BLEU score in Table 1. In Table 1, we observe that the smoothing trick helps to improve the translation performance in terms of BLEU score. We think that adding pseudo-count to estimate the probability of f_j given e_i helps to obtain a more accurate estimation for some rare phrase pairs whose probabilities are under-estimated. Since the provided dataset is relatively small for machine translation task, this smoothing trick helps in this case.

While the initialization trick does not seem to improve the performance. We conjecture that in the first iteration of the EM algorithm, the algorithm will also estimate θ_{f_j,e_i} based on the number of time that f_j and e_i appear in the corpus, which has similar effect as the initialization we try.

4 Conclusion

In this assignment, we develop the statistical machine translation by building the pipeline including word alignment, phrase extraction and WFST construction. We find that the statistical machine translation is good at memorizing rare phrase translation by the help of the phrase table. Even the training example appears once in the training corpus, it can almost sure to memorize it in the table. We further explore some tricks including, smoothing and initialization to boost the performance of the IBM Model 1. We find that the smoothing trick helps in the case where the training corpus is small.

References

[1] Cyril Allauzen, Michael Riley, Johan Schalkwyk, Wojciech Skut, and Mehryar Mohri. Openfst: A general and efficient weighted finite-state transducer library. In *International Conference on Implementation and Application of Automata*, pages 11–23. Springer, 2007.

[2] 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.

[3] Philipp Koehn, Franz Josef Och, and Daniel 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*, pages 48–54. Association for Computational Linguistics, 2003.

[4] Robert C Moore. Improving ibm word-alignment model 1. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 518. Association for Computational Linguistics, 2004.