Statistical Structure in Language Processing
Phrase based models

Cristina Gârbacea
10407936
cr1st1na.garbacea@gmail.com

Sara Veldhoen
10545298
sara.veldhoen@student.uva.nl

Abstract

Bla bla bla

1 Introduction

In this paper we explore the utility of phrase based models inside a statistical machine translation system. As compared to our previous assignment where we only used word aligned models, in this assignment our goal is to build an efficient phrase pair extraction tool that would extract phrase pairs of up to length 4 from a given word-aligned parallel training corpus.

Since word based models translate words as atomic units, they fall short of capturing dependencies between groups of words. Even more so in cases where each source word is aligned to exactly one target word. Phrase based models can overcome this limitation by treating phrases, sequences of words, as atomic translation units, in order to make use of local context in the translation process. Phrase based translation models give improved translations over the IBM models and are capable of giving state-of-the-art translations for many pairs of languages.

We implemented a phrase extraction algorithm that we compare to Moses, an existing statistical machine translation system that allows for automatically training translation models for any language pair. We introduce Moses and the evaluation metric we use, Bleu, in section 2. In section 3 follows a presentation of the phrase extraction algorithm we implemented and we offer an insight into how to get joint and conditional probability estimates. In section 4 we present the results we obtained and finally we conclude in section 5.

2 Moses and Bleu

3 Phrase Extraction and weight estimation

In this section, we present our approach to the extraction of phrase pairs from the corpus. We also compute coverage of the phrase table. The estimation of translation probabilities is done in both a conditional and joint fashion.

Phrase Extraction The number of possible phrase pairs per sentence pair is huge: each sentence can be partitioned in a vast amount of ways, and each partition could form a phrase pair with any partition in the paired sentence.

In order to reduce the space, we consider only phrase pairs that are consistent with word alignments produced by IBM models. As in (1), consistency is defined as follows:

$$\begin{aligned}\langle \bar{e}, \bar{f} \rangle \text{ is consistent with } A \Leftrightarrow \\ \forall e_i \in \bar{e} : \langle e_i, f_j \rangle \in A \Rightarrow f_j \in \bar{f}, \\ \text{and } \forall f_j \in \bar{f} : \langle e_i, f_j \rangle \in A \Rightarrow e_i \in \bar{e}, \\ \text{and } \exists e_i \in \bar{e}, f_j \in \bar{f} : \langle e_i, f_j \rangle \in A.\end{aligned}$$

For this assignment, the symmetrized alignments of the corpus sentences were given. We base our extraction algorithm on the one presented in (1, page 133). We iterate over all windows up to a certain length in the English sentence, and find the foreign windows that are consistent given the alignment. For all valid pairs of windows, we extract the corresponding phrase pair. Note that we keep counts of the thus extracted phrase pairs, for efficient weight estimation in a later stage.

Coverage of the phrase table To investigate the usefulness of the extracted phrases for unknown text, we compute the coverage as compared to held-out data: a similar but independent parallel

corpus. For this purpose, we run the same extraction algorithm on the held-out data. Then, we check for each extracted phrase pair whether it is in the phrase table. If it is not, however, we might be able to construct the phrase pair from phrases in the phrase table. This is a hard problem: each phrase in the held-out phrase pair of length n can be split in 2^{n-1} ways. Moreover, any phrase part in the source can be aligned to any phrase part in the target, thus introducing a combinatoric problem. Fortunately, we restricted the extracted phrases to a length of 4, so that the number of parts that we need to combine is at most $4 * 4$, with $4!$ possible combinations.

First, we create the possible splits for both the input and output phrase. Those are the input to the constructing algorithm. Since the phrase table is stored as a mapping from target to source phrases, we start from the target side. We begin with one possible target split, and look up the leftmost phrase part in the table. If it translates to any part of the foreign phrase, we recursively call the building algorithm with the remainder of the target and source phrase. Whenever a target part cannot be translated, the search for this split option is abandoned.

The coverage percentage is the relative number of phrase pairs in the heldout set that are either in the phrase table or can be constructed as described above.

Conditional Probability Estimates After having extracted the phrase pairs, we compute the conditional translation probability estimates for a foreign phrase $\overline{f}$ given an English phrase $\overline{e}$, using the following formula:

$$\phi(\overline{f}|\overline{e}) = \frac{count(\overline{e}, \overline{f})}{\sum_{\overline{f}_i} count(\overline{e}, \overline{f}_i)}$$

Here $count(\overline{e}, \overline{f})$ denotes in how many sentence pairs a specific phrase occurs and is extracted. To get relative frequency estimates, we normalize this value by the count of occurences of all phrase pairs containg the English phrase $\overline{e}$ inside the whole corpus.

Joint Probability Estimates In (2) quite a different approach is taken to phrase based translation. The idea of a noisy channel, that a foreign sentence is a corrupted version of an original English sentence, is abandoned. Rather, the two sentences are considered different substantiation of a bag of concepts. In this framework, the probability of a phrase pair is a joint probability conditioned on a concept. In practice, we do not explicitly model the concept but view the phrase pair itself as a concept, so its weight is just the joint translation probability of the two phrases: $t(\bar{e}, \bar{f}$.

The estimation of the translation probabilities in (2) is done in an adapted version of expectation maximization. In our implementation, we do not consider all alignments of all possible phrases, but instead base the extraction of phrases on the symmetrized word alignments from IBM models. Therefore, we can gather the counts of these phrase pairs directly:

$$\begin{aligned}\phi(\overline{f}, \overline{e}) &= \phi(\overline{f}|\overline{e}) \times \phi(\overline{e}) \\ &= \frac{count(\overline{e}, \overline{f})}{\sum_{\overline{f}_i, \overline{e}_i} count(\overline{e}, \overline{f}_i)}\end{aligned}$$

4 Experiments and Results

Phrase table coverage The training data consisted of 100,000 parallel sentences, the held-out corpus had 2000 sentences. Out of 95,073 consistent phrase pairs in the held-out corpus, $31,5\%$ could be constructed using the phrase table. Intuitively, this is quite a bad performance. As many as 95% of these were not directly extracted from the phrase table, but built from smaller parts.

We analyzed which phrase pairs would be built. We observed that the vast majority consisted of phrases with punctuation. Naturally, punctuation is often aligned in the symmetrized word alignments. The phrase extraction algorithm behaves greedily, in that it adds as much as possible to a consistent phrase pair, including punctuation. Although sequences of word can often contribute to meaningful phrase pairs, adding punctuation mainly appears to introduce a lot of data sparsity.

5 Conclusion

References

[1] Philipp Koehn, 2010. *Statistical Machine Translation*. Cambridge University Press.

[2] Daniel Marcu and William Wong, 2002. *A phrase-based, joint probability model for statistical machine translation*. Association for Computational Linguistics.

[3] Franz Josef Och, Christoph Tillmann, and Hermann Ney, 1999. *Improved alignment models for statistical machine translation* Proceedinfs of the Joint

SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora.