Phrase-based Symbolic Machine Translation Model

Xin Qian
xinq@andrew.cmu.edu

Abstract

This report describes the implementation detail and experimental analysis on the phrase-based machine translation model. Experiment was performed on the IWSLT2016 workshop for German-English translation. The baseline implementation achieved the BLEU scores of XXX and XXX (XXX improvement over the reference baseline) on validation and test set respectively.

1 Introduction

As an alternative approach to neural machine translation model, we apply symbolic model to the same translation task on German-English translation. Symbolic models relies heavily on counting (what being there in the training corpus) and is superior at remembering low-frequency information.

2 Dataset Specification and Statistics

Experiment was done over the dataset from the IWSLT2016 workshop for German-English translation. The dataset was divided into into three batches, including 90K parallel lines of German-English sentences as a training set, 887 lines as a validation set, and a 1.5K lines as a test set. An additional German blind test without English translation was also given. Each batch has at most three pre-processed versions, a NLTK tokenizer preprocessed tokenized version (.tok), a simple lowercased version(.low) and a sentence length filtered version (.filt). We use the lowercased filtered training version for training and test on the lowercased validation, testing and blind dataset.

3 Implementation Detail and Experimental Analysis

3.1 General Pipeline

The general pipeline divided into the training module and the test module. At the start of both module, training dataset was read to build a list of parallel token list pairs. For the version that allows null alignment, we append at the beginning of target sentence a NULL symbol with index 0.

In our baseline, alignment was done through EM algorithm on learning $P(e_{a_j}|f_j)$. This is slightly off to the suggested implementation on lecture note. We chose it expecting a one-to-many mapping between source language and target language. The parameter calculation processed was changed including initializing with the number of target vocabulary size, and subscription modifications with respect to equations as below,

$$\theta_{e,f} = c_{e,f}/c_f \tag{107}$$

$$P(f_j|e_t, F) = \frac{P(a_j = t|f_j, E)}{\sum_{j'=1}^{|F|} P(a_{j'} = t|f_{j'}, E)} \tag{110}$$

In this way, we found the one-best alignments that maximize $P(A|F, E)$ by finding the $argmax_t P(a_j = t|F, E)$. Our EM iterations were selected to be 8 and 30 respectively.

At phrase extraction stage, we follow the psudocode in Algorithm 6. There are two modifications, first, SP need not be a strict subset (but a subset) in line 9. Besides, when null alignment is not allowed, quasi-consecutiveness becomes strict consecutiveness and line 13 till line 18 are never executed.

During the phrase fst creation staage, we hashed all state values (phrase translation history) with additional placeholder indicating whether it spans both languages and the exact position of language

Table 1: BLEU on different variations

Variation	EM Iter	Valid	Test
Baseline 1	8	17.91	17.76
Baseline 2	**30**	**18.10**	**18.12**
With null alignment	8	15.87	16.24
Intersection	8	15.94	16.87
IBM2	30	17.63	17.49

being switch, therefore preventing several corner cases like wir, - we , versus wir - , we ,.

The given script then compile WFST into OpenFST standard coded models. We could choose to composite FST together for efficiency purpose (which will strongly decrease decoding time) or seperate ngram-fst and phrase-fst as two individual models. The decoding stage follows Section 12.4. The decoding time for baseline takes 35 ins on test dataset and 86 mins on blind dataset.

3.2 Null alignment

Despite the more intuitive nature of allowing null alignment solely in alignment stage, comparing table 1 with line 2 and 4, we see that without null alignment, the baseline is able to achieve a higher BLEU score thant allowing baseline. This might holds true since null alignment expands candidate phrase pool without eliminating noise (FP phrases.) It could be the case when bidirectional intersection and null alignment could work better together, unfortuately due to time limit, we were unable to explore this variation.

3.3 Intersection

We apply the same alignment bi-directionaly and select only the intersection of the two alignment set. The number of intersected alignments greatly decreases as compared to each direction. Comparing line 2 and 5 we see that intersection does not help much. It might be because intersection trunks the candidate phrase set greatly that although the extraction precision is high, recall is severely reduced.

3.4 Numbr of Iterations

Comparing Baseline 1 and 2 we see that increasing EM iteration numbers will not be disappointing. Each iteration takes 2 minutes on average to update the θ parameter.

3.5 IBM model 2

The modification from IBM model 1 to 2 includes another latent variable that needs to be updated in EM, we name it as β and calculated as below. Note that since we modify our alignment direction, this equation is different from equation 114. Equation 111 were also modified to factor in $P(j|a_j = t, |E|, |F|)$. Training on validation set for this improves on test set with a BLEU score increase of 14.7%. However, training on the training dataset shows an opposite trend, with a BLEU score 17.49 (decrease of 3.47%.)

$$\beta_{t,j,|F|,|E|} = c_{t,j,|F|,|E|} / c_{t,|F|,|E|} \tag{114}$$

$$P(j|a_j = t, |E|, |F|) = \frac{P(a_j = t|j, |E|, |F|)}{\sum_{j'=1}^{|F|} P(a_j = t|j, |E|, |F|)}$$
(Analogous to 110)

4 Future Work

4.1 Visualization

Due to time constraint and the Graphviz version support for mac OS beyond mountain lion, we did not visualize any of the ngram FST, phrase FST or the composited FST. However, it would be interesting to see some real FST figures instead of the toy example FSTs we have in notes to get more insight.

4.2 Efficiency Improvement

Decoding takes surprisingly longer time as the generated FST is large to search from. Efficiency could be improved by pruning generated phrases (ignoring any phrase that occurs only once) to reduce FST size. However, since phrase-based MT is advantageous to capture low frequency information, pruning could be risky to conteract the advantage point of phrase-based MT approach.