Statistical Machine Translation: Alignment and Training

Jasper Lin
Stanford University
jasperlin@stanford.edu

Vishnu Sundaresan
Stanford University
vishnu@stanford.edu

1 Introduction

This paper presents three different models used in producing word alignments from parallel text in an unsupervised manner. Additionally, we also investigate creating new features for the Phrasal system that will increase the Bilingual Evaluation Understudy (BLEU) score.

2 Word Alignment

2.1 Pointwise Mutual Information Model

The PMI model as a baseline and learning step was written in a way so that the code and structure could be reused and serve as a starting point for the IBM model implementations. We simply iterated through each of the source and target sentences, and counted the occurrences of each possible pair of words in order to determine a probability of two words being aligned. We also insert a null word into the target word vector so that each source word could potentially map to a null word, but did not iterate over it when outputting alignments. In the alignment function, we iterate through to find the target word with the best probability of being aligned to the current source word. On the test set, our PMI aligner performed with an AER of 0.622, much better than the expected AER range of 0.7 to 0.8.

2.2 IBM Model 1

The IBM model 1 implementation was based on the general structure of the PMI code, with the difference that the alignment outputs would be reversed. We would check the conditional probabilities of a certain french word being aligned to an english word, and take the maximum such alignment probability to output. For the learning aspect of the model, we iterate over the training data until the conditional probability model converges, as the parameters learn and change slowly. In order to prevent the learning process from looping infinitely, we set a hard iteration limit of 10, after testing and noticing that after 10 iterations the probabilities were changing only in the 4th decimal point onward, and would not be as large of a factor in calculating the best alignment.

Some problems and design decisions that we had to make during our implementation was how to store and access the summations for probabilities in order to normalize the counts and compute the delta to increment each parameter by. We restructured our code by swapping the order of the loops to first precompute the sum, then divide each parameter key pair by the sum in order to generate savings on runtime. Additionally, we did not realize that the null word appended to the end of the target sentence would perpetuate the the alignment function, and had to print out alignments to debug the source of our poor AER. Eventually, we optimize our model to achieve a test AER of 0.345 on the French-English dataset, which is also considerably better than the required 0.4.

2.3 IBM Model 2

IBM model 1 differs from model 1 in the general sense that it takes into account not only the conditional probabilities of alignments occurring, but the relative positions with respect to the overall sentence that the alignment occurs in. This helps to eliminate some of the errors seen in the model 1 example translation by putting more weight to alignments that occur along the diagonal of the grid. Additionally, the conditional alignment probabilities were initialized to be the values generated by model 1, giving us a place to iterate from. Like model 1, we set the convergence criteria to be 10 iterations as we found the probabilities converging to 4 decimal places. Many of the design choices we made for model 1 turned out to be useful in model 2 as well, with the simple addition of a q variable that counts the alignment positions of words in different sentence lengths.

2.4 Results

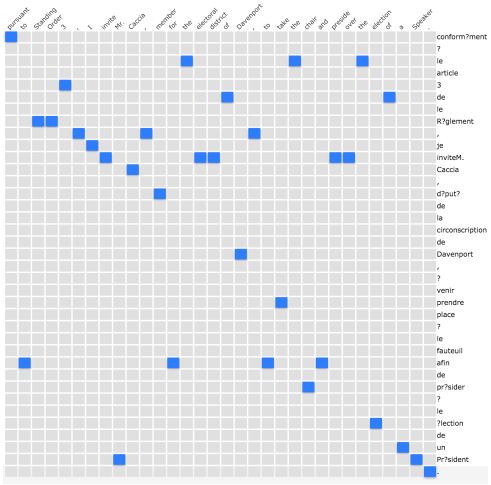

Figure 1: A sample word alignment using IBM Model 1

Figure 1 is shown a visualization of a sample alignment that our algorithm outputted. We can see that common words such as the and I always map to the correct French words le and je. Unfortunately, uncommon words such as electoral and preside do not have strong probabilities in the learned parameters, and are instead mapped to a much more commonly occurring french word invit. Additionally, the grid shows a rough alignment along the diagonal, but not to the degree that we should expect for a long sentence.

Figure 2 is the visualization of the alignment be-

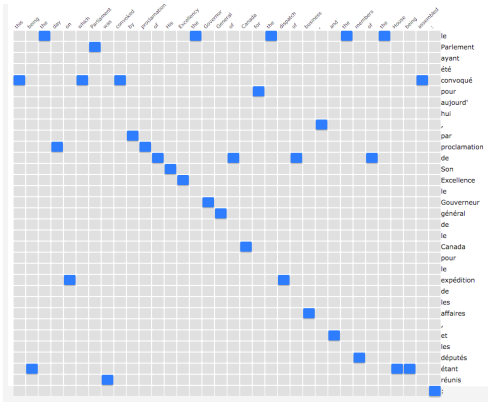

Figure 2: A sample word alignment using IBM Model 2

tween a French and English sentence. Unlike IBM 1, we can see that the outputted alignment lies for the majority along the diagonal. One drawback that we can see is that we map the earliest occurring French word to the English word. If words appear multiple times in the French sentence, the alignments will technically be wrong even though the correct word is actually translated. Additionally, we can see that certain words are incorrectly translated such as assembled to convoqu when it should translate to runis. After going through different examples, towards the end of the sentence the alignments seem to become more and more incorrect and deviate from the diagonal. Since the counts for the parameter q are based on the word length comparisons, sentences of longer lengths are underrepresented and have less data. Even though the conditional probabilities are very high, the overall probability is offset by the low q. In contrast, higher values for q that result from smaller training sentence pairs inflate the low conditional probabilities. This is an unfortunate drawback that model 1, which saw a higher error rate than the AER of 0.314 of model 2, does not have as it only takes into account the conditional probabilities.

	F-E	H-E	C-E
PMI	0.686	0.838	0.845
Model 1	0.361	0.598	0.593
Model 2	0.3292	0.607	0.591

Table 1: The French data had 10000 training sentences and an evaluation set size of 37. The Hindi data had 3441 training sentences and an evaluation set size of 25 sentences. The Chinese data had 10000 sentences and an evaluation set size of 2818 sentences.

	F-E	H-E	C-E
PMI	0.622	0.828	0.852
Model 1	0.345	0.580	0.6179
Model 2	0.314	0.601	0.608

Table 2: AER result table on test set. The French data had 10000 training sentences and an evaluation set size of 111. The Hindi data had 3441 training sentences and an evaluation set size of 22 sentences. The Chinese data had 10000 sentences and an evaluation set size of 977 sentences.

3 Phrasal Feature

In order to supplement the baseline Phrasal system, we experimented with a couple different features to be incorporated into the log-linear model.

One feature we explored was the difference between the number of words in the source sentence and the number of words in the target sentence. This feature gives more information to the model about how similar the languages are. This feature directly relates to the concepts of alignments; it allows the model to better understand one-to-one alignments, many-to-one, one-to-many, and NULL mappings by supplying this data. On average, this feature resulted in an increase of in BLEU. One drawback of this feature is that longer sentences might naturally have a higher difference in sentence length, but our feature simply captures the difference. If one result of the feature is to use more or fewer NULL words based on the difference, then longer sentences would have an inflated number of NULL translations as the ratio is not examined. This feature had an average BLEU of 15.42, about 0.13 better than the average baseline performance provided. This feature probably works significantly better for languages that share a similar sentence structuring. We would expect the impact of this feature to be significantly greater for an English to French translation in comparison to an English to Chinese translation. It was an interesting to compare the results of the translation with this feature to the baseline translation. Because we are applying weights to the difference in lengths, the model seems to favor one to one mappings. There are more functional words inserted into the translation as a result.For example, here is a translation of the sentence il aurait fallu 226 voix pour l approuver to English via the baseline system: it would take 226 votes to approve it' and here is the translation via our system incorporating our new feature: it would have taken 226 votes to approve it. This distinction is subtle but requires the system to have knowledge of different verb conjugations, which can result in the many-to-one mappings. Our feature seemingly gives the model more flexibility to add in functional words to better express different conjugations, and as such seems to serve its intended purpose. One other feature we tried was determining if the French sentence ended in a question mark. Both English and the French language have the common characteristic of a set number of words that indicate the start or structure of a question. Unfortunately, French has multiple words that may map to each English question token such as "quoi", "que", and "quel" mapping to "what". Each of these is chosen depending on the context of the sentence, thus adding an indicator feature for questions should increase the weights of words that appear with specific question tokens. This feature had an average BLEU of 15.52, about 0.23 better than the average baseline performance provided.

4 Conclusion

We successfully implemented, tested, and analyzed three separate yet related algorithms for translation and alignment. These algorithms take a very naive approach to understanding how words might align to each other and the result of this simplified approach is reflected in the results. We also explored using different features to supplement and improve the Phrasal translation. We found that provide the system with more information about the structure of the sentence is a good way to improve results. Machine translation is an extremely nontrivial task and top translation software utilize much more in-depth understanding of sentence structure in grammar to achieve better results.

References

Michael Collins 2011. *Statistical Machine Translation: IBM Models 1 and 2*

Kevin Knight 2009. *A Statistical MT Tutorial Workbook*