Structural Patterns in Translation

Cynthia Day, Caroline Ellison
CS 229, Machine Learning
Stanford University
cyndia, cellison

Introduction

Our project seeks to analyze word alignments between translated texts. The motivation for this study was the inversion transduction grammar proposed by Dekai Wu [6]. It models the alignments between bilingual sentence pairs through the use of parse trees that represent the alignments as rearrangements of phrases between the two translations. Ultimately, we hope to bring about a better understanding of word rearrangements in translation, which could be used to improve automated translators.

Background

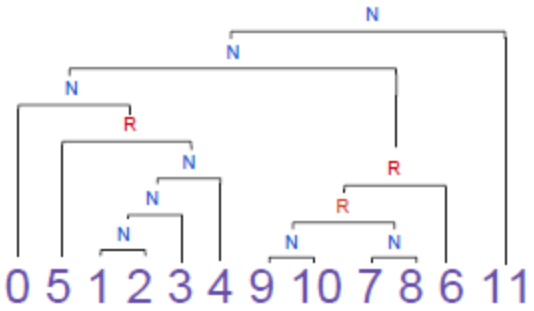

Fig 1. A standard parse tree

Dekai Wu states that the differences in grammar between any two sentences can be described by a set of operations on pairs of phrases, represented by nodes in a tree. He describes a method of taking alignment data, represented by a string of numbers indicating the position of the word in the translated sentence (e.g. 4 3 2 5 1), and performing an operation that combines two adjacent phrases into a larger one by either concatenating them or transposing their order in the sentence. His idea was to try to recreate the word order in the original string (represented by 1 2 3 4 5) by repeatedly performing these operations. The algorithm is initialized by treating each word in the sentence as a separate node. These nodes form the leaves of the tree. In the example alignment described above (4 3 2 5 1), the first two nodes would be combined in reverse order to give R(3,4) 2 5 1, where (a,b) refers to the interval spanned by a and b. A second "reverse" operation can be performed to produce R(2,4) 5 1, followed by a "normal" concatenation and a "reverse" one to obtain the original word order. The aggregate node formed by combining two smaller nodes is made the parent of the latter nodes, and the process of concatenation and transposition results in the generation of a

parse tree. We have pictured above a more complicated potential parse tree.

Data

We analyzed data taken from the Europarl Corpus [2], which consists of the proceedings of the European Parliament and their translations into the various official European languages. We utilized the language pairs German-English, French-English, and Spanish-English. The word alignments of these translations were derived using automated software provided by the NAACL 2006 workshop on statistical machine translation. The software indexed the words in the original text and matched them with the corresponding indices of the words in the translation.

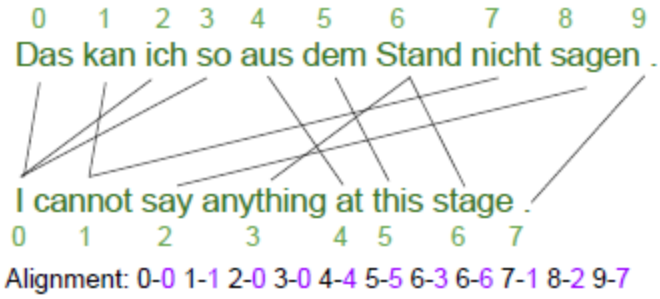

Fig 2. A sample word alignment between a German sentence and its English translation.

In general, we used 10,000 lines of each corpus as training data and drew 1,000 lines from a different section to use as testing data.

Determining Direction of Translation

We first built a classifier that, given raw word alignment data, determined the direction of translation. The classifier read automatically generated word-alignment data one line at a time, where each line of the data corresponded to the word alignment for one sentence. Each line was read both forwards and backwards, so that we had data for both English-foreign and foreign-English word alignments. We then put the forwards and backwards data into three-dimensional arrays. Specifically, we stored frequency counts for each word alignment, which we represented by index in the English sentence, index in the foreign sentence, and the alignment length (since a word often maps to multiple words in the second language). We used Naive Bayes to determine the probabilities of any given word alignment resulting from each language pair and used the probabilities to classify that word alignment, for the following results.

Language Pair	Accuracy
English-German	0.648
English-Spanish	0.616

English-French	0.733

Naive Bayes works under the assumption that features are independent of each other. This assumption is not obviously justified in the case of word alignment, since rearrangements of words in a sentence can have dependencies on other words. Support vector machines make no assumptions about independence and often get better results than Naive Bayes algorithms, so we decided to test the performance of SVMs on our data using the LibSVM library[1]. We used the possible word alignments as our features, so that the feature vectors for each sentence had entries of 0 for unused word alignments and 1 for used word alignments. We tested on C-SVC and nu-SVC paired with radial basis function, sigmoid, and polynomial kernels, and found that both runtime and accuracy rate were on overall worse than when we used Naive Bayes. For example, on C-SVC with a radial basis function as the kernel, we obtained the following results.

Language Pair	Accuracy
English-German	0.6040
English-Spanish	0.6880
English-French	0.5865

Since different parts of speech will rearrange in distinct ways, we decided to improve our classifier by incorporating an automated part-of-speech tagger provided by the Stanford Natural Language Processing Group [4], [5]. We were able to mark the part of speech of each word alignment. We then used a four-dimensional array to store frequency counts, where the part-of-speech tag was used as an additional dimension. As can be seen below, adding parts of speech significantly improved our classification accuracy.

Language Pair	Accuracy
English-German	0.847
English-Spanish	0.882
English-French	0.766

Classifying Language Pairs

Beyond classifying direction of translation, we decided to utilize the different languages represented in the data to build a classifier that, given word alignment data, classified it into one of two or three language pairs. Since our data always involved the translation of English into a foreign language, we trained our classifier to identify the foreign language; the language set for a classifier is the set of potential foreign languages. We used the

same Naive Bayes algorithm that was used to classify direction of translation, including part-of-speech tagging because of the increased accuracy it brings. We obtained the following results.

Language Set	Accuracy
German/Spanish	0.687
German/French	0.668
Spanish/French	0.629
German/Spanish/French	0.515

*Note that for language sets of size two, random guessing would have expected accuracy 0.5, while for language sets of size three, random guessing would have expected accuracy 0.333. Thus, our algorithm does significantly better than random guessing.

Given different language pairs, one would expect that their parse trees would have distinct characteristics, and that knowledge of these characteristics could be used to improve translation. By incorporating inversion transduction grammar parse trees into the classifier, we hoped to gain some understanding of the extent that parse trees differ between languages.

We used the nodes of the parse trees generated for each sentence alignment, recording whether they were “normal” or “reverse” and storing these counts for each tree. We then implemented the classifier using a Naive Bayes algorithm.

Language Set	Accuracy
German/Spanish	0.523
German/French	0.528
Spanish/French	0.559
German/Spanish/French	0.364

This gave significantly worse results than the classifier that did not rely on binary trees. This was unexpected, since we hypothesized that as a more linguistically natural way to express word rearrangements, binary trees would give better results. However, it appears that the tree structures for each language do not differ much in the above language pairs.

Conclusion

We focused on two distinct goals--classifying direction of translation, and classifying into language pairs. We found that Naive Bayes provided similar results and but was far more computationally efficient than SVMs, so we used Naive Bayes for the majority of our project. Using part of speech tagging, we were able to get good accuracy for both of our classification objectives, but analysis of parse trees was surprisingly unhelpful.

Further Study

An area left to explore is the accuracy of our algorithm on non-European language data. We hypothesize that with greater structural differences between languages, accuracy increases significantly. However, such a test would be accurate only if all the translations were based off the same original text. In particular, when we attempted to incorporate a separate Arabic-English parallel corpus [3] into our language set, we obtained extremely skewed results, with virtually 100% accuracy on Arabic. However, upon closer examination, it was clear that this was at least partially due to structural differences between the English texts chosen to be translated, thus we decided to discard the results.

Acknowledgments

We would like to thank Professor Martin Kay for his suggestion of the project and his support throughout it. In addition, we would like to thank Jia-Han Chiam and Vishesh Gupta for providing the code to generate parse trees and contributing some background to this report, including the word alignment diagrams featured.

References

[1] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1--27:27, 2011.
[2] Phillip Koehn. Europarl: A multilingual corpus for evaluation of machine translation. MT Summit 2005.
[3] Jörg Tiedemann, 2012, Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the 8th International Conference on Language Resources and Evaluation* (LREC 2012)
[4] Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of HLT-NAACL 2003*, pp. 252-259.
[5] Kristina Toutanova and Christopher D. Manning. 2000. Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*, pp. 63-70.
[6] Dekai Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics,* 23(3):377-403, September 1997.