Spectral Clustering for Example Based Machine Translation

Rashmi Gangadharaiah
LTI
Carnegie Mellon University
Pittsburgh P.A. 15213
rgangadh@andrew.cmu.edu

Ralf Brown
LTI
Carnegie Mellon University
Pittsburgh P.A. 15213
ralf@cs.cmu.edu

Jaime Carbonell
LTI
Carnegie Mellon University
Pittsburgh P.A. 15213
jgc@cs.cmu.edu

Abstract

Prior work has shown that generalization of data in an Example Based Machine Translation (EBMT) system, reduces the amount of pre-translated text required to achieve a certain level of accuracy (Brown, 2000). Several word clustering algorithms have been suggested to perform these generalizations, such as k-Means clustering or Group Average Clustering. The hypothesis is that better contextual clustering can lead to better translation accuracy with limited training data. In this paper, we use a form of spectral clustering to cluster words, and this is shown to result in as much as 29.08% improvement over the baseline EBMT system.

1 Introduction

In EBMT, the source sentence to be translated is matched against the source language sentences present in a corpus of source-target sentence pairs. When a partial match is found, the corresponding target translations are obtained through subsentential alignment. These partial matches are put together to obtain the final translation by optimizing translation and alignment scores and using a statistical target language model in the decoding process. Prior work has shown that EBMT requires large amounts of data (in the order of two to three million words) (Brown, 2000) of pre-translated text, to function reasonably well. Thus, some modification of the basic EBMT method is required to make it effective when less data is available. In order to use the available text efficiently, systems such as, (Veale and Way, 1997) and (Brown, 1999), convert the examples in the corpus into templates against which the new text can be matched. Thus, source-target sentence pairs are converted to source-target generalized template pairs. An example of such a pair is shown below:

The session opened at 2p.m
La séance est ouverte á 2 heures

The <event> <verb-past-tense> at <time>
La <event> <verb-past-tense> a <time>

This single template can be used to translate different source sentences, including for example,

The session adjourned at 6p.m
The seminar opened at 8a.m

if 'session' and 'seminar' are both generalized to '<event>', 'opened' and 'adjourned' are both generalized to '<verb-past-tense>' and finally '6p.m' and '8a.m' are both generalized to '<time>'.

The system used by (Brown, 1999) performs its generalization using both equivalence classes of words and a production rule grammar. This paper describes the use of spectral clustering (Ng. et. al., 2001; Zelnik-Manor and Perona, 2004), for automated extraction of equivalence classes. Spectral clustering is seen to be superior to Group Average Clustering (GAC) (Brown, 2000) both in terms of semantic similarity of words falling in a single cluster, and overall BLEU score (Papineni. et. al., 2002) in a large scale EBMT system.

The next section explains the term vectors extracted for each word, which are then used to cluster words into equivalence classes and provides an outline of the Standard GAC algorithm. Section 3 describes the spectral clustering algorithm used. Sec-

Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL, pages 41–44,
New York, June 2006.

tion 4 lists results obtained in a full evaluation of the algorithm. Section 5 concludes and discusses directions for future work.

2 Term vectors for clustering

Using a bilingual dictionary, usually created using statistical methods such as those of (Brown et. al., 1990) or (Brown, 1997), and the parallel text, a rough mapping between source and target words can be created. This word pair is then treated as an indivisible token for future processing. For each such word pair we then accumulate counts for each token in the surrounding context of its occurrences (N words, currently 3, immediately prior to and N words immediately following). The counts are weighted with respect to distance from occurrence, with a linear decay (from 1 to 1/N) to give greatest importance to the words immediately adjacent to the word pair being examined. These counts form a pseudo-document for each pair, which are then converted into term vectors for clustering.

In this paper, we compare our algorithm against the incremental GAC algorithm(Brown, 2000). This method examines each word pair in turn, computing a similarity measure to every existing cluster. If the best similarity measure is above a predetermined threshold, the new word is placed in the corresponding cluster, otherwise a new cluster is created if the maximum number of clusters has not yet been reached.

3 Spectral clustering

Spectral clustering is a general term used to describe a group of algorithms that cluster points using the eigenvalues of 'distance matrices' obtained from data. In our case, the algorithm described in (Ng. et. al., 2001) was performed with certain variations that were proposed by (Zelnik-Manor and Perona, 2004) to compute the scaling factors automatically and for the k-Means orthogonal treatment (Verma and Meila, 2003) during the initialization. These scaling factors help in self-tuning distances between points according to the local statistics of the neighborhoods of the points. The algorithm is briefly described below.

1. Let S =$s_1, s_2, ...s_n$, denote the term vectors to be clustered into k classes.
2. Form the affinity matrix A defined by
   $A_{ij} = exp(-d^2(s_i, s_j)/\sigma_i\sigma_j)$ for $i \neq j$
   $A_{ii} = 1$
   Where, $d(s_i, s_j) = 1/(sim(s_i, s_j) + \epsilon)$
   $sim(s_i, s_j)$ is the Cosine similarity between s_i and s_j, ϵ is used to prevent the ratio from becoming infinity
   σ_i is the set of local scaling parameters for s_i. σ_i = $d(s_i, s_T)$ where, s_T is the T^{th} neighbor of point s_i for some fixed T (7 for this paper).
3. Define D to be the diagonal matrix given by, $D_{ii} = \Sigma_j A_{ij}$
4. Compute $L = D^{-1/2}AD^{-1/2}$
5. Select k eigenvectors corresponding to k largest eigenvalues (k is presently an externally set parameter). The eigenvectors are normalized to have unit length. Form matrix U by stacking all the eigenvectors in columns.
6. Form the matrix Y by normalizing U's rows, $Y_{ij} = U_{ij}/\sqrt{(\Sigma_j U_{ij}^2)}$
7. Perform k-Means clustering treating each row of Y as a point in k dimensions. The k-Means algorithm is initialized either with random centers or with orthogonal vectors.
8. After clustering, assign the point s_i to cluster c if the corresponding row i of the matrix Y was assigned to cluster c.
9. Sum the distances between the members and the centroid of each cluster to obtain the classification cost.
10. Goto step 7, iterate for a fixed number of iterations. In this paper, 20 iterations were performed with orthogonal k-Means initialization and 5 iterations with random k-Means initialization.
11. The clusters obtained from the iteration with least classification cost are selected as the k clusters.

4 Preliminary Results

The clusters obtained from the spectral clustering method are seen by inspection to correspond to more natural and intuitive word classes than those obtained by GAC. Even though this is subjective and not guaranteed to lead to improve translation performance, it shows that maybe the increased power of spectral clustering to represent non-convex classes

(non-convex in the term vector domain) could be useful in a real translation experiment. Some example classes are shown in Table 1. The first class in an intuitive sense corresponds to measurement units. We see that in the <units> case, GAC misses some of the members which are actually distributed among many different classes and hence these are not well generalized. In the second class <months>, spectral clustering has primarily the months in a single class whereas GAC adds a number of seemingly unrelated words to the cluster. The classes were all obtained by finding 80 clusters in a 20,000-sentence pair subset of the IBM Hansard Corpus (Linguistic Data Consortium, 1997) for spectral clustering. 80 was chosen as the number of clusters since it gave the highest BLEU score in the evaluation. For GAC, 300 clusters were used as this gave the best performance.

To show the effectiveness of the clustering methods in an actual evaluation, we set up the following experiment for an English to French translation task on the Hansard corpus. The training data consists of three sets of size 10,000 (set1), 20,000 (set2) and 30,000 (set3) sentence pairs chosen from the first six files of the Hansard Corpus. Only sentences of length 5 to 21 words were taken. Only words with frequency of occurrence greater than 9 were chosen for clustering because more contextual information would be available when the word occurs frequently and this would help in obtaining better clusters. The test data was chosen to be a set of 500 sentences obtained from files 20, 40, 60 and 80 of the Hansard corpus with 125 sentences from each file. Each of the methods was run with different number of clusters and results are reported only for the optimal number of clusters in each case.

The results in Table 2 show that spectral clustering requires moderate amounts of data to get a large improvement. For small amounts of data it is slightly worse than GAC, but neither gives much improvement over the baseline. For larger amounts of data, again both methods are very similar, though spectral clustering is better. Finally, for moderate amounts of data, when generalization is the most useful, spectral clustering gives a significant improvement over the baseline as well as over GAC. By looking at the clusters obtained with varying amounts of data, it can be concluded that high pu-

Table 1: Clusters for <units> and <months>

Spectral clustering	GAC
“adjourned” “hre” “cent” “%” “days” “jours” “families” “familles” “hours” “heures” “million” “millions” “minutes” “minutes” “o’clock” “heures” “p.m.” “heures” “p.m.” “hre” “people” “personnes” “per” “%” “times” “fois” “years” “ans”	“adjourned” “hre” “families” “familles” “million” “millions” “o’clock” “heures” “p.m.” “heures” “people” “personnes” “per” “%” “times” “fois”
“august” “août” “december” “décembre” “february” “février” “january” “janvier” “march” “mars” “may” “mai” “november” “novembre” “october” “octobre” “only” “seulement” “june” “juin” “july” “juillet” “april” “avril” “september” “septembre” “since” “depuis”	“august” “août” “december” “décembre” “february” “février” “january” “janvier” “march” “mars” “may” “mai” “november” “novembre” “october” “octobre” “only” “seulement” “june” “juin” “july” “juillet” “april” “avril” “september” “septembre” “page” “page” “per” “$” “recognize” “parole” “recognized” “parole” “recorded” “page” “section” “article” “since” “depuis” “took” “séance” “under” “loi”

Table 2: % Relative improvement over baseline EBMT
clus is the number of clusters for best performance

	GAC		Spectral	
	% Rel imp	#clus	% Rel imp	#clus
10k	3.33	50	1.37	20
20k	22.47	300	29.08	80
30k	2.88	300	3.88	200

rity clusters can be obtained with even just moderate amounts of data.

5 Conclusions and future work

From the experimental results we see that spectral clustering leads to relatively purer and more intuitive clusters. These clusters result in an improved BLEU score in comparison with the clusters obtained through GAC. GAC can only collect clusters in convex regions in the term vector space, while spectral clustering is not limited in this regard. The ability of spectral clustering to represent non-convex shapes arises due to the projection onto the eigenvectors as described in (Ng. et. al., 2001).

As future work, we would like to analyze the variation in performance as the amount of data increases. It is widely known that increasing the amount of training data in a generalized EBMT system eventually leads to saturation of performance, where all clustering methods perform about as well as baseline. Thus, all methods have an operating region where they are the most useful. We would like to locate and extend this region for spectral clustering.

Also, it would be interesting to compare the clusters obtained with spectral clustering and the Part of Speech tags of the words in the same cluster, especially for languages such as English where good taggers are available.

Finally, an important direction of research is in automatically selecting the number of clusters for the clustering algorithm. To do this, we could use information from the eigenvalues or the distribution of points in the clusters.

Acknowledgment

This work was funded by National Business Center award NBCHC050082.

References

Andrew Ng, Michael Jordan, and Yair Weiss 2001. On Spectral Clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14: Proceeding of the 2001 Conference,* pages 849-856, Vancouver, British Columbia, Canada, December.

Deepak Verma and Marina Meila. 2003. Comparison of Spectral Clustering Algorithms. `http://www.ms.washington.edu/~spectral/`.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei Jing Zhu. 2002. BLEU: a method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002),* pages 311-318,Philadelphia, PA, July. `http://acl.ldc.upenn.edu/P/P02`

Linguistic Data Consortium. 1997. *Hansard Corpus of Parallel English and French.* Linguistic Data Consortium, December. `http://www.ldc.upenn.edu/`

L. Zelnik-Manor and P. Perona 2004 Self-Tuning Spectral Clustering. In *Advances in Neural Information Processing Systems 17: Proceeding of the 2004 Conference.*

Peter Brown, J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, J. Lafferty, R. Mercer and P. Roossin. 1990. A Statistical Approach to Machine Translation. *Computational Linguistics,* 16:79-85.

Ralf D. Brown. 1997. Automated Dictionary Extraction for "Knowledge-Free" Example-Based Translation. In *Proceedings of the Seventh International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-97),* pages 111-118, Santa Fe, New Mexico, July. `http://www.cs.cmu.edu/~ralf/papers.html`

Ralf D. Brown. 1999. Adding Linguistic Knowledge to a Lexical Example-Based Translation System. In *Proceedings of the Eighth International Conference on Theoretical and Methodological Issues in Machine Translation(TMI-99),* pages 22-32, August. `http://www.cs.cmu.edu/~ralf/papers.html`

Ralf. D. Brown. 2000. Automated Generalization of Translation Examples. In *Proceedings of Eighteenth International Conference on Computational Linguistics (COLING-2000),* pages 125-131, Saarbrücken, Germany.

Tony Veale and Andy Way. 1997. Gaijin: A Template-Driven Bootstrapping Approach to Example-Based Machine Translation. In *Proceedings of NeMNLP97, New Methods in Natural Language Processing,* Sofia, Bulgaria, September. `http://www.compapp.dcu.ie/~tonyv/papers/gaijin.html`.