Cluster Stopping Rules for Word Sense Discrimination

Guergana Savova, Terry Therneau and Christopher Chute,
Mayo Clinic, Rochester, MN, USA
[Savova.Guergana;therneau;chute]@mayo.edu

Abstract

As text data becomes plentiful, unsupervised methods for Word Sense Disambiguation (WSD) become more viable. A problem encountered in applying WSD methods is finding the exact number of senses an ambiguity has in a training corpus collected in an automated manner. That number is not known a priori; rather it needs to be determined based on the data itself. We address that problem using cluster stopping methods. Such techniques have not previously applied to WSD. We implement the methods of Calinski and Harabasz (1975) and Hartigan (1975) and our adaptation of the Gap statistic (Tibshirani, Walter and Hastie, 2001). For evaluation, we use the WSD Test Set from the National Library of Medicine, whose sense inventory is the Unified Medical Language System. The best accuracy for selecting the correct number of clusters is 0.60 with the C&H method. Our error analysis shows that the cluster stopping methods make finer-grained sense distinctions by creating additional clusters. The highest F-scores (82.89), indicative of the quality of cluster membership assignment, are comparable to the baseline majority sense (82.63) and point to a path towards accuracy improvement via additional cluster pruning. The importance and significance of the current work is in applying cluster stopping rules to WSD.

1 Introduction

The dominant approach in word sense disambiguation (WSD) is based on supervised learning from manually sense-tagged text. While this is effective, it is quite difficult to get a sufficient number of manually sense-tagged examples to train a system. Mihalcea (2003) estimates that 80-person years of annotation would be needed to create training corpora for 20,000 ambiguous English words, given 500 instances per word. For that reason, we are developing unsupervised knowledge-lean methods that avoid the bottlenecks created by sense-tagged text. Unsupervised clustering methods utilize only raw corpora as their source of information, and there are growing amounts of general and specialized domain corpora available, e.g. biomedical domain corpora.

Improvements in WSD methods would be of immediate value in indexing and retrievals of biomedical text given the explosion of biomedical literature as well as the rapid deployment of electronic medical records. Semantic/conceptual indexing and retrieval in that domain is often done in regard to the Unified Medical Language System (UMLS) developed at the National Library of Medicine (NLM) at the United States National Institutes of Health (NIH)[1]. It is important to understand that the UMLS is significantly different than a dictionary, which is often the source of the sense inventory. Rather, the UMLS integrates more than 100 medical domain controlled vocabularies such as SNOMED-CT[2] and the International Classification of Diseases (ICD)[3]. UMLS has three main components. The first component, the Metathesaurus, includes all terms from the controlled vocabularies and is organized by concept, which is a cluster of terms representing the same meaning. Each concept is assigned a concept unique identifier (CUI), which is inherited by each term in the cluster. UMLS-based semantic indexing is based on CUI assignments. The second component, the Semantic Network, groups the concepts into 134 types of categories and indicates the relationships between them. The Semantic Network is a coarse ontology of the concepts. The third component, the SPECIALIST lexicon, contains syntactic information for the Metathesaurus terms.

[1] http://www.nlm.nih.gov/pubs/factsheets/umls.html
[2] http://www.snomed.org/
[3] http://www.who.int/classifications/help/icdfaq/en/

MeSH, an ontology within UMLS, is heavily used for indexing biomedical scientific publications, e.g. Medline[4]. Hospitals, medical practices and biomedical research increasingly rely on the UMLS, or a subset ontology within it, to index and retrieve relevant information. It is estimated that approximately 7400 UMLS terms map to multiple concepts which creates ambiguity (Weeber, Mork and Aronson., 2001). Term ambiguity has been pointed out to be one of the major challenges for UMLS-based semantic indexing and retrieval (Weeber et al., 2001). For example, "cold" has the following six UMLS meanings, each with its own UMLS CUI: cold temperature, common cold, cold sensation, cold therapy, chronic obstructive lung disease (COLD), and Cold brand of chlorpheniramine-phenylpropanolamine.

The problem we are addressing in this paper is discovering the number of senses an ambiguous word has in a given corpus, which is a component within a completely unsupervised WSD system. For example, if a corpus of 1000 instances containing the word "cold" has been compiled from patients medical records, how many "cold" senses are in that corpus? This is a challenge any NLP system implementing WSD faces. To address this problem, we apply cluster stopping rules in an automated way.

The paper is organized as follows. Section 2 overviews the related work on cluster stopping rules. Section 3 outlines our methods, tools, features selection, test set and evaluation metrics. Section 4 presents the results and discusses them. Section 5 is the conclusions.

2 Background and Related Work

Our work is based on cluster analysis. Cluster analysis is often performed to discover the groups that the data naturally fall into. The number of groups is not known a priori; rather, it needs to be determined based on the data itself. Such methods or "cluster stopping rules" usually rely on within-cluster dissimilarity/error (*W(k)*) metrics which in general exhibit a decline when the number of clusters increases. Splitting a natural group into subgroups reduces the criterion less than when well-separated clusters are discovered. In those cases, the *W(k)* will not have a sharp decline as the instances are close. This phenomenon has been described in statistical literature as the "elbow" effect as illustrated in Figure 1. Methods for locating the "elbow have been the goal of many research studies (Hartigan, 1975; Calinski and Harabasz, 1975; Milligan and Cooper, 1987; Tibshirani, Walter and Hastie, 2001 among many).

Milligan and Cooper (1985) offer the most comprehensive comparative study of the performance of 30 stopping rules. They carry out their study on "mildly truncated data from multivariate normal distributions, and one would not expect their ranking of the set of stopping rules to be reproduced exactly if a different cluster-generating strategy were adopted." (Gordon, 1999, p. 61). The five rules which were the top performance in the Milligan and Cooper study are Calinski and Harabasz (1974) a.k.a. C&H, Goodman and Kruskal (1954), C index (Hubert and Schultz, 1976), Duda and Hart (1973) and Beale (1969). Tibshirani et al. (2001) introduce the Gap statistic and compare its performance to the methods of Calinski and Harabasz (1974), Krzanowski and Lai (1985), Hartigan (1975), and the Silhouette method (Kaufman and Rousseeuw, 1990). On the simulated and DNA microarray data Tibshirani and colleagues used for their experiments, the Gap statistic yields the best result.

In general, stopping rules fall into two categories – global and local (Gordon, 1999; Tibshirani et al., 2001). *Global rules* take into account a combination of within-cluster and between-cluster similarity measures over the entire data. Global rules choose such k where that combined metric is optimal. Global rules, however, in most cases do not work for k=1, that is they do not make predictions of when the data should not be partitioned at all. Global rules look at the entire data over k number of clusters. *Local rules*, on the other hand, are based only on a given k solution or individual pairs of clusters and test whether they should be grouped together. They need a threshold value or a significance level, which depends on the specific data and in most cases have to be empirically determined.

3 Methodology

3.1 Overview

In this study, we explore three cluster stopping methods as applied to unsupervised WSD – Hartigan (1975), Calinski and Harabasz (1974), and the Gap statistic (Tibshirani et al., 2001). The data to be clustered is instances of context surrounding each ambiguity. Each instance is converted into a feature vector where the features are

[4] http://www.nlm.nih.gov/pubs/factsheets/medline.html

ngrams (unigrams or bigrams) and each cell is the frequency of occurrence of a unigram or bigram or the log-likelihood of a bigram occurring in that particular instance after applying a feature selection method. The clustering algorithm for this set of experiments is agglomerative clustering (see Section 3.5 for a more detailed description).

Our goal is to group contexts into separate clusters based on the underlying sense of the ambiguous word. Thus, the observations are contexts and the features are the identified lexical features (i.e. significant word(s)) that represent the contexts. Our observed data matrix generally shows the following characteristics –

1) it is discrete
2) it is high dimensional/multivariate
3) it can be real valued or integer, or binary
4) it is sparse; while the number of features can be in few hundreds, contexts have a length limit (ignoring the commonly occurring "closed class words" like "the", "an", "on" etc.)
5) it represents a distribution of contexts that is generally skewed.

Following is our motivation for choosing the three cluster stopping rules. Hartigan (1975) and Calinski and Harabasz (1974) have been consistently used as baselines in a number of studies, e.g. Tibshirani et al. (2001). The Hartigan method is computationally simple and efficient and unlike C&H, it is defined for k=1. The C&H method was ranked the top among 30 stopping rules in the comprehensive study conducted by Milligan and Cooper (1975). The Gap statistic (Tibshirani et al., 2001) is a fairly recent method that has gained popularity by showing excellent results when applied to the bio domain, e.g. clustering DNA mircoarray data. None of the methods, however, have been applied or adapted to WSD.

3.2 Calinski and Harabasz (1975) Method

The C&H method is reported to perform the best among 30 stopping rules (Milligan and Cooper, 1985). C&H is a global method. The Variance Ratio Criteria C&H uses is

$$VRC(k) = \frac{\frac{BGSS(k)}{k-1}}{\frac{WGSS(k)}{n-k}}$$

where *BGSS* (between group sum of squares) is the sum of the dispersions between the k cluster centroids and the general centroid; *WGSS* (within-group sum of squares) is the sum of each cluster's dispersion of its cluster members (measured by the sum of squared distances between each member and the cluster centroid) weighed by the number of cluster members; *k* is the number of clusters and *n* is the number of instances. The distance used is the Euclidean distance. As Calinski and Harabasz point out, *VRC* "is analogous to the F-statistic in univariate analysis" (Calinski and Harabasz, 1975, p. 10). C&H seeks to maximize *VRC*.

3.3 Hartigan (1975) Method

Hartigan (1975) proposes a cluster stopping rule:

$$H(k) = \left[\frac{WGSS(k)}{WGSS(k+1)} - 1 \right](n - k - 1)$$

where *n* is the total number of instances to be clustered, *k* is the number of clusters and *WGSS(k)* is the total sum of squared distances of cluster members from their cluster centroid in all clusters when clustered in k clusters.

H(k) is used to decide when *k+1* clusters are needed rather than *k* clusters. Its distribution approximates the F distribution. A large value of *H(k)* would indicate that the addition of a cluster is warranted. Hartigan suggests that as a crude rule of thumb, values exceeding 10 justify increasing the number of clusters from k to k+1 (Hartigan, 1975, p. 91). Thus, a solution is the smallest $k \geq 1$ such that $H(k) \leq 10$. The method can return 1 cluster as the optimal solution. Hartigan (1975) is a local method.

3.4 Gap Statistic Method

In general, the "gap" method compares the difference/gap between the within-cluster dispersion measure for the observed distribution and that for an appropriate null distribution of the data. Tibshirani and colleagues (Tibshirani et al., 2001) start with the assumption of a single cluster null model which is to be rejected in favor of a k-component model (k>1) if the observed data supports it. Tibshirani and colleagues use a uniform distribution as the null distribution of the data to standardize the comparison between all the *W(k)* over the various values of k where *W(k)* is the pooled within cluster sum of squares around the cluster means (distance is squared Euclidean distance). The uniform distribution is the least favorable distribution and the most likely to produce spurious clusters. However,

Tibshirani and colleagues also point out that the choice of an appropriate null distribution depends on the data. Tibshirani and colleagues compare the curve of *log(W(k))* to the *log(W*(k))* curve obtained from the reference uniformly distributed over the data. The estimated optimal number of clusters is the k value where the gap between the two curves is the largest. Figure 1 is an example of *log(W(k))* to the *log(W*(k))* curves used in the computation of the Gap statistic.

The two main advantages of the Gap statistic over various previously proposed "stopping rules" are its ability to work with data created by almost any type of clustering and its ability to accurately estimate the optimal number of clusters even for data that naturally falls into just one cluster. The Gap statistic is an application of parametric bootstrap methods to the clustering problem. Unlike non-parametric methods, parametric techniques represent the observed data distribution. The basic strategy is to create multiple random data sets over the observed distribution for which there are no clusters, apply the chosen clustering method to them, and tabulate the apparent decrease in within-cluster variation that ensues. This gives a measure of optimism with which to compare the clustering of the observed data.

The complete methodology can be broadly classified into two important components namely the reference distribution and the algorithm which uses the reference distribution. We describe each of the two components below.

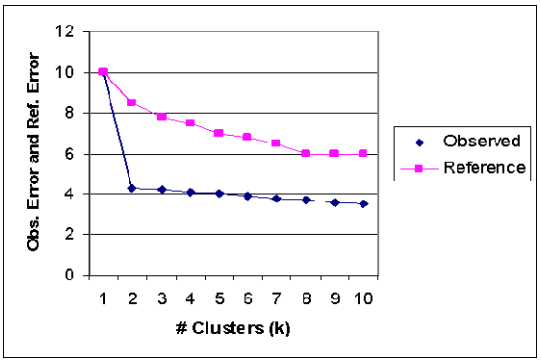

Figure 1: The functions log(W(k)) (observed) and log(W*(k)) (reference) used for computing the Gap statistic

Reference Distribution Generation for an NLP Task

Here, we describe how we extend the generation of the reference distribution over the observed data to retain the characteristics mentioned at end of section 3.1. We will use the observed data shown in Table 1 as a running example. To simulate the structure of the observed data, the following features are to be emulated:

(a) Context length is the number of features that can occur in a context. Contexts can be sentences, paragraphs, entire documents or just any specified window size. In general, the number of available features will be at least in the hundreds, however, only a few might occur in a given context, especially if the context is limited to the sentence the target ambiguity occurs in. Additionally, context length is influenced by the feature selection method – if only very frequent lexical units are retained as features, then only those units will represent the context. Thus, a context length could be very small compared to the size of the feature set. In the example from Table 1, context length is captured by the row marginals, e.g. the context length for Context1 is 3, which means that overall there are only three features for that context.

(b) Sparsity is a consequence of relatively small context length. Currently, our assumption is that contexts are derived from small discourse units (sentences or abstracts at the most). For bigger discourse units, e.g. several paragraphs or entire documents, our proposed generation of the reference distribution should be modified to reflect feature occurrences over those units. In the example from Table 1, for instance in Context1, there are 3 features that are present – Feature1, Feature4 and Feature5 – the rest are absent. Sparsity can be viewed as the number of absent/zero-valued features for each row.

(c) Feature distribution is the frequency of occurrence of each feature across all contexts. It is captured by the column marginals of the observed data matrix. For example, in Table 1 Feature1 occurs twice over the entire data; similarly Feature2 occurs twice and so on. Feature distribution can be viewed as the number of occurrences of each feature in the entire corpus.

Now we describe how we do the reference generation to stay faithful to the characteristics described above. We use the *uniform* and the *proportional* methods. The *uniform* method generates data that realizes (a) and (b) characteristics of the data and is the used originally in Tibshirani et al. (2001). The *proportional* method captures (a), (b), and (c) and is our adaptation of the Gap method.

The data is constructed as follows. To retain the context lengths of the observed data in the

	Feature1	Feature2	Feature3	Feature4	Feature5	…FeatureP	Total number of non-zero value cells
Context1	1	0	0	1	1	……	3
Context2	0	1	1	0	1	……	3
Context3	1	0	0	0	1	……	2
Context4	0	1	1	1	1	……	4
…ContextN	……	……	……	……	……	……	……
	2	2	2	2	4	……	12

Table 1: Observed data (sample)

reference data, the row marginals of the reference data are fixed to be equal to those of the observed data. In Table 1, the row marginals for the reference data will be {3, 3, 2, 4}. Carrying the observed marginals to the reference data applies to both the *uniform* and *proportional* methods. Note that currently we fix only the row marginals. Due to the current assumption of binary feature frequency, the generated reference data is binary too and this is true for both methods.

The main difference between the uniform and proportional methods lies in whether the feature distribution is maintained in the simulation. The *uniform* method does not weigh the features; rather, all features are given equal probability of occurring in the generated data. A uniform random number r over the range [1, featureSetSize] is drawn. The cell corresponding to the r^{th} column (i.e. feature) in the current row under consideration (i.e. context) is assigned "1". For example, in our running example let's say we are generating reference data for the 3rd row from Table 1. We first generate a random number over the range [1, p]. Let's assume that the generated number is 4. Then, the cell [3, 4] is assigned value "1". This procedure is repeated twice since the row marginal for this row of the reference data is 2. The *proportional* method factors in the distribution of the column marginals of the observed data while generating the random data. Unlike the uniform method, it takes into account the weight of each feature. In other words, the features by their frequency assign themselves a range. For example, the features in the Table 1 will be assigned the following ranges: Feature1 - [1, 2]; Feature2 - [3, 4]; Feature3 – [5, 6]; Feature4 – [7, 8]; Feature5 – [9, 12]. A random number is generated over the range [1, total number of feature occurrences]. For the data in Table 1, a random number is generated over the range [1, 12]. The feature corresponding to the range in which the random number falls is assigned "1". For example, if we are generating the reference for Context3 and the generated random number over the range [1, 12] is 5, then a look-up determines that 5 falls in the range for Feature3. Hence, the cell in Context3 corresponding to Feature3 is assigned "1". Similar to the *uniform* method we would repeat this procedure twice to achieve the row marginal total of 2.

Currently we proceed with the binary reference data created by the procedure described above. Note that this binary reference matrix can be converted to a strength-of-association matrix by multiplying it with a diagonal matrix that contains the strength-of-association scores, e.g. log likelihood ratio, Mutual Information, Pointwise mutual information, Chi-squared to name a few.

Algorithm

The complete algorithm of the Gap Statistics which the reference distribution is a part of is:

1. Cluster the observed data, varying the total number of clusters from k = 1, 2, …., K, giving within dispersion measures W(k), k = 1, 2, ….K.
2. Generate B reference datasets using the *uniform* or the *proportional* methods as described above, and cluster each one giving within dispersion measures W*(kb), b = 1, 2, … B, k = 1, 2,… K. Compute the estimated Gap statistic:

$$Gap(k) = (1/B)\sum_b \log(W^*(kb)) - \log(W(k))$$

3. Let $\bar{l} = (1/B)\sum_B \log(W^*(kb))$, compute the standard deviation $sd(k) = [(1/B)\sum_B (\log(W^*(kb) - \bar{l})^2]^{1/2}$ and define $s(k) = sd(k)\sqrt{1+1/B}$. Finally choose the number of clusters via

$$\hat{k} = smallestKsuchThat\_Gap(k) \geq Gap(k+1) - s(k+1)$$

The final step is the criterion for selecting the optimal k value. It says to choose the smallest k value for which the gap is greater than the gap for the earlier k value by the significance test of "one standard error". The "one standard error" calculations are modified to account for the simulation error. Tibshirani and colleagues also advise to use a multiplier to the s(k) for better rejection of the null hypothesis.

3.5 Tools, Feature Selection and Method Parameters

For feature representation, selection, context representation and clustering, we used SenseClusters0.69 (http://senseclusters.sourceforge.net). It offers a variety of lexical features (ngrams, collocations, etc.) and feature selection methods (frequency, log likelihood, etc.). The contexts can then be represented with those features in vector space using first or second order vectors which are then clustered. A detailed description can be found in Purandare and Pedersen (2004) and http://www.d.umn.edu/~tpederse/senseclusters.html. SenseClusters links to CLUTO for the clustering part (http://www-users.cs.umn.edu/~karypis/cluto/download.html). CLUTO implements in a fast and efficient way the main clustering algorithms – agglomerative, partitional and repeated bisections.

We chose the following methods for feature representation and selection. *Method1* uses bigrams as features, average link clustering in similarity space and the abstract as the context to derive the features from. The method is described in Purandare and Pedersen (2004). It is based on first order context vectors, which represent features that occur in that context. A similarity matrix is clustered using the average link agglomerative method. Purandare and Pedersen (2004) report that this method generally performed better where there was a reasonably large amount of data available (i.e., several thousand contexts). The application of that method to the biomedical domain is described in a technical report (Savova, Pedersen, Kulkarni and Purandare, 2005). *Method2* uses unigrams which occur at least 5 times in the corpus. The context is the abstract. The choice of those features is motivated by Joshi, Pedersen and Maclin (2005) study which achieves best results with unigram features.

For the Hartigan cluster stopping method, the threshold is set to 10 which is the recommendation in the original algorithm. For the Gap cluster stopping method, we experiment with B=100, and the uniform and proportional reference generation methods.

3.6 Test Set

Our test set is the NLM WSD[5] set which comprises 5000 disambiguated instances for 50 highly frequent ambiguous UMLS Metathesaurus strings (Weeber et al., 2001). Each ambiguity has 100 manually sense-tagged instances. All instances were randomly chosen from Medline abstracts. Each ambiguity instance is provided with the sentence it occurred in and the Medline abstract text it was derived from. The senses for every ambiguity are the UMLS senses plus a "none of the above" category which captures all instances not fitting the available UMLS senses.

For the current study, we modified the NLM WSD by excluding instances sense-tagged with the "none of the above" category. This is motivated by the fact that that category is a catch-all category for all senses that do not fit the current UMLS inventory. First, we excluded words whose majority category was "none of the above". Secondly, from the instances of the remaining words, we removed those marked with "none of the above". That subset of the original NLM WSD set we refer to as the "modified NLM WSD set" (Table 2).

3.7 Evaluation

Our evaluation of the performance of the cluster stopping rules is two-fold. *Accuracy* is a direct evaluation measuring the correctly recognized number of senses:

$$\frac{\textit{words with correctly predicted number of senses}}{\textit{all words}}$$

Accuracy evaluates how well the methods discover the exact number of senses in the test corpus. The *F-score* of the WSD is an indirect evaluation for the quality of the cluster assignment:

$$F\_score = \frac{(\beta^2 + 1)\,\mathrm{Precision}\,\mathrm{Recall}}{\beta^2\,\mathrm{Precision} + \mathrm{Recall}}$$

Precision is the number of correctly clustered instances divided by the number of clustered instances; *Recall* is the number of correctly clustered instances divided by all instances. There may be some number of contexts that the clustering algorithm declines to process, which leads to the difference in precision and recall.
Our baseline is a simple clustering algorithm that assigns all instances of a target word to a single cluster.

4 Results and Discussion

Table 3 presents the results for the three methods

[5]http://wsd.nlm.nih.gov/Restricted/Reviewed_Results/index.shtml

Word, instances, senses after removal of "none of the above" sense	*Word, instances, senses after removal of "none of the above" sense*	*Word, instances, senses after removal of "none of the above" sense*
Adjustment, 93, 3	Frequency, 94, 1	Radiation, 98, 2
Blood pressure, 100, 3	Growth, 100, 2	Repair, 68, 2
Cold, 95, 4	Immunosuppression, 100, 2	Scale, 65, 1
Condition, 92, 2	Implantation, 98, 2	Secretion, 100, 1
Culture, 100, 2	Inhibition, 99, 2	Sex, 100, 3
Degree, 65, 2	Japanese, 79, 2	Single, 100, 2
Depression, 85, 1	Ganglion, 100, 2	Strains, 93, 2
Determination, 79, 1	Glucose, 100, 2	Surgery, 100, 2
Discharge, 75, 2	Man, 92, 4	Transient, 100, 2
Energy, 100, 2	Mole, 84, 2	Transport, 94, 2
Evaluation, 100, 2	Mosaic, 97, 2	Ultrasound, 100, 2
Extraction, 87, 2	Nutrition, 89, 4	Variation, 100, 2
Fat, 73, 2	Pathology, 99, 2	White, 90, 2
Fluid, 100, 1	Pressure, 96, 1	

Table 2: Modified NLM WSD set

In terms of accuracy (Table 3, column 3), the C&H method has the best results ($p<0.01$ with t-test). Note that the modified NLM WSD set contains seven words with one sense – depression, pressure, determination, fluid, frequency, scale, secretion – for which the C&H method is at a disadvantage as it cannot return one cluster solution.

In terms of predicted number of senses (Table 3, column 5), the Hartigan method tends to underestimate the number of senses (overcluster), thus making coarser sense distinctions. The adapted Gap and C&H methods tend to overestimate them (undercluster), thus making finer grained sense distinctions.

In terms of cluster member assignment as demonstrated by the F-scores (Table 3, column 4), our adapted Gap method and the Hartigan method perform better than the C&H method ($p<0.05$ with t-test). The Hartigan method F-scores along with Gap uniform with Method 1 feature selection are not significantly different from the baseline ($p>0.05$ with t-test); the rest are significantly lower than the majority sense baseline ($p<0.05$ with t-test).

The high F-scores point to a path for improving accuracy results. Singleton clusters could be pruned as they could be insignificant to sense discrimination. As it was pointed out, the best performing algorithms (C&H and Gap proportional) tend to create too many clusters (Table 3, column 5). Another way of dealing with singleton or smaller clusters is to present them for human review as they might represent new sense distinctions not included in the sense inventory.

One explanation for the performance of the stopping rules (overclustering in particular) might be that some senses are very similar, e.g. "cold temperature" and "cold sensation" for the "cold" ambiguity in instances like "Her feet are cold." Another explanation is that the stopping rules rely on the clustering algorithm used. In our current study, the experiments were run with only agglomerative clustering as implemented in CLUTO. The distance measure that we used is Euclidean distance, which is only one of many choices. Yet another explanation is in the feature sets we experimented with. They performed very similarly on both the accuracy and F-scores. Future work we plan to do is aimed at experimenting with different features, clustering algorithms, distance measures as well as applying Singular Value Decomposition (SVD) to the reference distribution matrix for our adapted Gap method. We are actively pursuing reference generation with fixed column and row marginals. The work of Pedersen, Kayaalp and Bruce (1996) uses this technique to find significant lexical relationships. They use the CoCo (Badsberg, 1995) package which implements the Patefield (1981) algorithm for I x J tables. Another venue is in the combination of several stopping rules which will take advantage of each rule's strengths. Yet another component that needs to be addressed towards the path of completely automated WSD is cluster labeling.

5 Conclusions

In this work, we explored the problem of discovering the number of the senses in a given target ambiguity corpus by studying three cluster stopping rules. We implemented the original algorithms of Calinski and Harabasz (1975) and Hartigan (1975) and adapted the reference generation of the Gap algorithm (Tibshirani et al., 2001) to our task. The best accuracy for selecting the correct number of clusters is 0.60 with the

Feature Selection	Stopping Rule	Accuracy	F-score (baseline majority sense = 82.63)	Average number of senses (true average number of senses = 2.19)
Method1	C&H	0.49	80.71	2.90 (overestimates)
	Hartigan	0.10	82.15	1.27 (underestimates)
	Gap (uniform)	0.02	82.00	1.49 (underestimates)
	Gap (proportional)	0.24	81.31	2.51 (overestimates)
Method2	C&H	0.60	80.27	3.36 (overestimates)
	Hartigan	0.02	82.89	1.10 (underestimates)
	Gap (uniform)	0.05	81.63	2.44 (overestimates)
	Gap (proportional)	0.12	81.15	2.59 (overestimates)

Table 3: Results – accuracy, F-score and predicted average number of sense

C&H method. Our error analysis shows that the cluster stopping methods make finer-grained sense distinctions by creating additional singleton clusters. The F-scores, indicative of the quality of cluster membership assignment, are in the 80's and point to a path towards accuracy improvement via additional cluster pruning.

Acknowledgements

The Perl modules of our implementations of the algorithms can be downloaded from http://search.cpan.org/dist/Statistics-CalinskiHarabasz/, http://search.cpan.org/dist/Statistics-Hartigan/, http://search.cpan.org/dist/Statistics-Gap/. We are greatly indebted to Anagha Kulkarni and Ted Pedersen for their participation in this research. We would also like to thank Patrick Duffy, James Buntrock and Philip Ogren for their support and collegial feedback, and the Mayo Clinic for funding the work.

References

A. D. Gordon. 1999. Classification (Second Edition). Chapman & Hall, London

A. Purandare and T. Pedersen. 2004. Word Sense Discrimination by Clustering Contexts in Vector and Similarity Spaces. Proceedings of the Conference on Computational Natural Language Learning (CoNLL): 41-48, May 6-7, 2004, Boston, MA

E. M.L. Beale. 1969. Euclidean cluster analysis. Bulletin of the International Statistical Institute:92–94, 1969

G. Savova, T. Pedersen, A. Kulkarni and A. Purandare. 2005. Resolving Ambiguities in the Biomedical Domain. Technical Report. Minnesota Supercomputing Institute.

G. W. Milligan and M.C. Cooper. 1985. An examination of procedures for determining the number of clusters in a data set. Psychometrika 50:159-179.

J. H. Badsberg. 1995. An environment for graphical models. PhD dissertation, Aalborg University.

J. Hartigan. 1975. Clustering Algorithms, Wiley, New York.

L. A. Goodman and W.H. Kruskal. 1954. Measures of association for cross classifications. J. of Amer. Stat. Assoc., 49:732--764, 1954.

L. Hubert and J. Schultz. 1976. Quadratic assignment as a general data-analysis strategy. British Journal of Mathematical and Statistical Psychologie. 29:190-241

L. Kaufman and P. Rowsseeuw. 1990. Finding groups in data: an introduction to cluster analysis. New York. Wiley.

M. Joshi, T. Pedersen and R. Maclin. 2005. A comparative study of support vector machines applied to the supervised word sense disambiguation problem in the medical domain. IICAI. India.

M. Weeber, J. Mork and A. Aronson. 2001. Developing a test collection for biomedical word sense disambiguation. Proc. AMIA

R. B. Calinski and J. Harabasz. 1974. A dendrite method for cluster analysis. Communications in statistics 3:1-27.

R. Mihalcea. 2003. The role of non-ambiguous words in natural language disambiguation. RANLP-2003, Borovetz, Bulgaria

R. O. Duda and P. E. Hart. 1973. Pattern Classification and Scene Analysis. Wiley, New York, 1973

R. Tibshirani, G. Walther and T. Hastie. 2001. Estimating the number of clusters in a dataset via the Gap statistic. Journal of the Royal Statistics Society (Series B).

T. Pedersen, M. Kayaalp and R. Bruce. 1996. Significant lexical relationships. Proc. of the 13th National Conference on Artificial Intelligence, August 1996, Portland, Oregon.

W. J. Krzanowski and Y. T. Lai. 1985. A criterion for determining the number of groups in a data set using the sum of squares clustering. Biometrics 44:23-34.

W. Patefield. 1981. An efficient method of generating random R x C tables with given row and column totals. Applied Statistics 30:91-97.