Towards Unsupervised Extraction of Verb Paradigms from Large Corpora

Cornelia H. Parkes, Alexander M. Malek and Mitchell P. Marcus
Department of Computer & Information Science
University of Pennsylvania
cparkes@linc.cis.upenn.edu*

Abstract

A verb paradigm is a set of inflectional categories for a single verb lemma. To obtain verb paradigms we extracted left and right bigrams for the 400 most frequent verbs from over 100 million words of text, calculated the Kullback Leibler distance for each pair of verbs for left and right contexts separately, and ran a hierarchical clustering algorithm for each context. Our new method for finding unsupervised cut points in the cluster trees produced results that compared favorably with results obtained using supervised methods, such as gain ratio, a revised gain ratio and number of correctly classified items. Left context clusters correspond to inflectional categories, and right context clusters correspond to verb lemmas. For our test data, 91.5% of the verbs are correctly classified for inflectional category, 74.7% are correctly classified for lemma, and the correct joint classification for lemma and inflectional category was obtained for 67.5% of the verbs. These results are derived only from distributional information without use of morphological information.

1 Introduction

This paper presents a new, largely unsupervised method which, given a list of verbs from a corpus, will simultaneously classify the verbs by lemma and inflectional category. Our long term research goal is to take a corpus in an unanalyzed language and to extract a grammar for the language in a matter of hours using statistical methods with minimum input from a native speaker. Unsupervised methods avoid labor intensive annotation required to produce the training materials for supervised methods. The cost of annotated data becomes particularly onerous for large projects across many languages, such as machine translation. If our method ports well to other languages, it could be used as a way of automatically creating a morphological analysis tool for verbs in languages where verb inflections have not already been thoroughly studied. Precursors to this work include (Pereira et al, 1993), (Brown et al. 1992), (Brill & Kapur, 1993), (Jelinek, 1990), and (Brill et al, 1990) and, as applied to child language acquisition, (Finch & Chater, 1992).

Clustering algorithms have been previously shown to work fairly well for the classification of words into syntactic and semantic classes (Brown et al. 1992), but determining the optimum number of classes for a hierarchical cluster tree is an ongoing difficult problem, particularly without prior knowledge of the item classification. For semantic classifications, the correct assignment of items to classes is usually not known in advance. In these cases only an unsupervised method which has no prior knowledge of the item classification can be applied. Our approach is to evaluate our new, largely unsupervised method in a domain for which the correct classification of the items is well known, namely the inflectional category and lemma of a verb. This allows us to compare the classification produced by the unsupervised method to the classifications produced by supervised methods. The supervised methods we examine are based on information content and number of items correctly classified. Our unsupervised method uses a single parameter, the expected size of the cluster. The classifications by inflectional category and lemma are additionally interesting because they produce trees with very different shapes.

* This work was supported by grants from Palladium Systems and the Glidden Company to the first author. The comments and suggestions of Martha Palmer, Hoa Trang Dang, Adwait Ratnaparkhi, Bill Woods, Lyle Ungar, and anonymous reviewers are also gratefully acknowledged.

The classification tree for inflectional category has a few large clusters, while the tree for verb lemmas has many small clusters. Our unsupervised method not only performs as well as the supervised methods, but is also more robust for different shapes of the classification tree.

Our results are based solely on distributional criteria and are independent of morphology. We completely ignore relations between words that are derived from spelling. We assume that any difference in form indicates a different item and have not "cleaned up" the data by removing capitalization, etc. Morphology is important for the classification of verbs, and it may well solve the problem for regular verbs. However, morphological analysis will certainly not handle highly irregular, high frequency verbs. What is surprising is that strictly local context can make a significant contribution to the classification of both regular and irregular verbs. Distributional information is most easily extracted for high frequency verbs, which are the verbs that tend to have irregular morphology.

This work is important because it develops a methodology for analyzing distributional information in a domain that is well known. This methodology can then be applied with some confidence to other domains for which the correct classification of the items is not known in advance, for example to the problem of semantic classification.

2 Data

This report is on our investigation of English text using verbs tagged for inflectional category.[1] The tags identify six inflectional categories: past tense (VBD), tenseless (VB), third-person singular present tense (VBZ), other present tense (VBP), -ing (VBG) and participle (VBN). The use of tagged verbs enables us to postpone the problem of resolving ambiguous inflectional forms, such as the homonyms, "work" as tenseless and "work" as present tense, a conflation that is pervasive in English for these categories. We also do not address how to separate the past participle and passive participle uses of VBN.

The methods reported in this paper were developed on two different corpora. The first corpus consisted of the 300 most frequent verbs from the 52 million word corpus of the New York Times, 1995.[2] For this corpus, both the verbs and the contexts consisted of tagged words. As a somewhat independent test, we applied our methods to the 400 most frequent verbs from a second corpus containing over 100 million words from the Wall Street Journal (1990-94). For the second corpus, the tags for context words were removed. The results for the two corpora are very similar. For reasons of space, only the results from the second corpus are reported here.

Table 1: Distribution of verbs by inflectional category in 400 most frequent verbs

Inflectional Category	Verbs
VB	109
VBN	80
VBD	76
VBG	67
VBZ	46
VBP	22

The distribution of verbs is very different for inflectional category and lemma. The distribution of verbs with respect to lemmas is typical of the distribution of tokens in a corpus. Of the 176 lemmas represented in the 400 most frequent verbs, 79 (45%) have only one verb. One lemma, BE, has 14 verbs.[3] Even in 100 million words, the 400^{th} most frequent verb occurs only 356 times. We have not yet looked at the relation between corpus frequency and clustering behavior of an item. The distribution of verbs in inflectional categories has a different profile (See Table 1). This may be related to the fact that, unlike lemmas, inflectional categories form a small, closed class.

3 Cluster Analysis

For each verb we extracted frequency counts for left and right bigrams called the left and right contexts, respectively. A similarity matrix for left and right contexts was created by calculating the relative entropy or Kullback Leibler (KL) distance [4] between the vectors of context frequency counts for each pair of verbs. The KL distance has been used previously (Magerman & Marcus (1990), Pereira et al. (1993))

[1] The verbs were automatically tagged by the Brill tagger. Tag errors, such as "| \$VBG" tended to form isolated clusters.

[2] Tagged corpora were provided by the Linguistic Data Consortium (http://www.ldc.upenn.edu).

[3] Upper and lower case forms are counted as distinct.

[4] For an introduction to relative entropy see (Cover & Thomas, 1991)

to measure the similarity between two distributions of word bigrams. For the moment we added a small constant to smooth over zero frequencies. Because the distance between $verb_i$ and $verb_j$ is not in general equal to the distance between $verb_j$ and $verb_i$, the KL distances between each pair of verbs are added to produce a symmetric matrix. We tested other measures of distance between words. The total divergence to the average, based on the KL distance (Dagan et al, 1994), produced comparable results, but the the cosine measure (Schuetze, 1993) produced significantly poorer results. We conclude that entropy methods produce more reliable estimates of the probability distributions for sparse data (Ratnaparkhi, 1997).

The similarity matrices for left and right contexts are analyzed by a hierarchical clustering algorithm for compact clusters. The use of a "hard" instead of a "soft" clustering algorithm is justified by the observation (Pinker, 1984) that the verbs do not belong to more than one inflectional category or lemma.[5] A hierarchical clustering algorithm (Seber, 1984) constructs from the bottom up using an agglomerative method that proceeds by a series of successive fusions of the N objects into clusters. The compactness of the resulting cluster is used as the primary criterion for membership. This method of complete linkage, also known as farthest neighbor, defines the distance between two clusters in terms of the largest dissimilarity between a member of cluster L_1 and a member of cluster L_2. We determined experimentally on the development corpus that this algorithm produced the best clusters for our data.

Figures 1 and 2 show portions of the cluster trees for left and right contexts. The scales at the top of the Figures indicate the height at which the cluster is attached to the tree in the arbitrary units of the distance metric. The left context tree in Figure 1 shows large, nearly pure clusters for the VBD and VBZ inflectional categories. The right context tree in Figure 2 has smaller clusters for regular and irregular verb lemmas. Note that some, but not all, of the forms of BE form a single cluster.

To turn the cluster tree into a classification, we need to determine the height at which to terminate the clustering process. A cut point is a

[5]The only exception in our data is " 's" which belongs to the lemmas BE and HAVE.

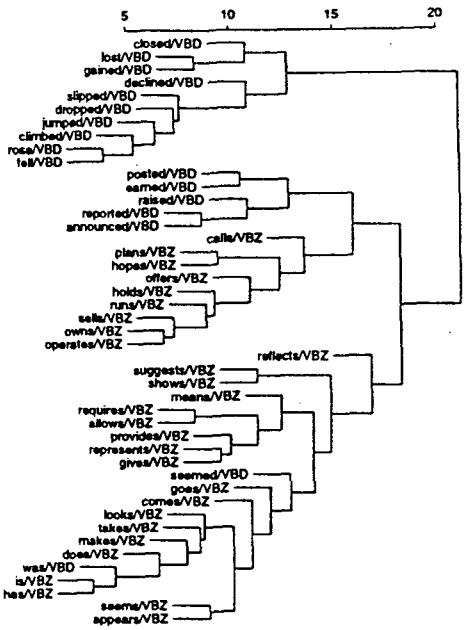

Figure 1: Verb clusters for left contexts

line drawn across the tree at height T that defines the cluster membership. A high cut point will produce larger clusters that may include more than one category. A low cut point will produce more single category clusters, but more items will remain unclustered. Selecting the optimum height at which to make the cut is known as the cutting problem.

A supervised method for cutting the cluster tree is one that takes into account the known classification of the items. We look at supervised methods for cutting the tree in order to evaluate the success of our proposed unsupervised method.

4 Supervised methods

For supervised methods the distribution of categories C in clusters L at a given height T of the tree is represented by the notation in Table 2. For a given cluster tree, the total number of categories C, the distribution of items in categories n_c, and the total number of items N are constant across heights. Only the values for L, m_l, and f_{cl} will vary for each height T.

There are several methods for choosing a

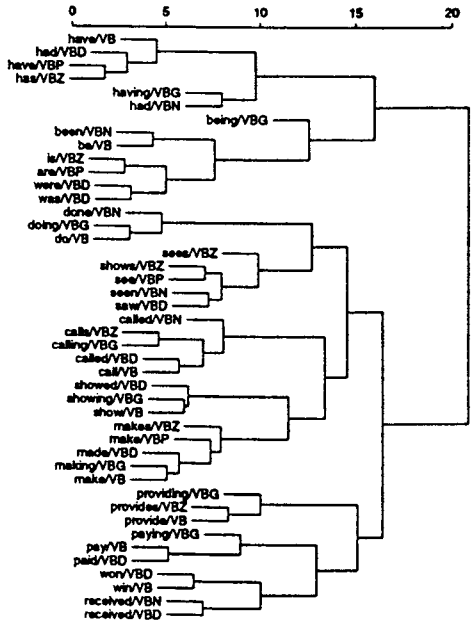

Figure 2: Verb clusters for right contexts

	clus-1	...	clus-L	Total
cat-1	f_{11}	...	f_{1L}	n_1
...	...	...	...	...
cat-C	f_{C1}	...	f_{CL}	n_C
Total	m_1	...	m_L	N

C is the number of categories
L is the number of clusters
m_l is the number of instances in cluster l
f_{cl} is the instances of category c in cluster l
N is the total number of instances for cut T
n_c is the number of instances in category c

Table 2: Notation

cut point in a hierarchical cluster analysis.[6] We investigated three supervised methods, two based on information content, and a third based on counting the number of correctly classified items.

4.1 Gain Ratio

Information gain (Russell & Norvig, 1995) and gain ratio (Quinlan, 1990) were developed as metrics for automatic learning of decision trees. Their application to the cutting problem for cluster analysis is straightforward. Information gain is a measure of mutual information, the reduction in uncertainty of a random variable given another random variable (Cover & Thomas, 1991). Let C be a random variable describing categories and L another random variable describing clusters with probability mass functions $p(c)$ and $q(l)$, respectively. The entropy for categories $H(C)$ is defined by

$$H(C) = -\sum_c p(c) log_2 p(c)$$

where $p(c) = n_c/N$ in the notation of Table 2. The average entropy of the categories within clusters, which is the conditional entropy of categories given clusters, is defined by

$$H(C|L) = -\sum_l q(l) \sum_c p(c|l) log_2 p(c|l)$$

where $q(l) = m_l/N$ and $p(c|l) = f_{cl}/m_l$ in our notation.

Information gain and mutual information $I(C;L)$ are defined by

$$\text{gain} = I(C;L) = H(C) - H(C|L)$$

Information gain increases as the mixture of categories in a cluster decreases. The purer the cluster, the greater the gain. If we measure information gain for each height T, $T = 1, ..., 40$ of the cluster tree, the optimum cut is at the height with the maximum information gain.

Information gain, however, cannot be used directly for our application, because, as is well known, the gain function favors many small clusters, such as found at the bottom of a hierarchical cluster tree. Quinlan (1990) proposed the gain ratio to correct for this. Let $H(L)$ be the entropy for clusters defined, as above, by

$$H(L) = -\sum_l q(l) log_2 q(l)$$

Then the gain ratio is defined by

$$\text{gain ratio} = \frac{I(C;L)}{H(L)}$$

The gain ratio corrects the mutual information between categories and clusters by the entropy of the clusters.

[6] For a review of these methods, see (Seber, 1984).

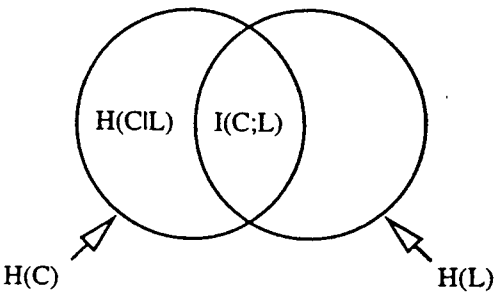

Figure 3: Relationship between entropy and mutual information adapted from (Cover & Thomas, 1991).

4.2 Revised Gain Ratio

We found that the gain ratio still sometimes maximizes at the lowest height in the tree, thus failing to indicate an optimum cut point. We experimented with the revised version of the gain ratio, shown below, that sometimes overcomes this difficulty.

$$\text{revised ratio} = \frac{I(C;L)}{H(L) - H(C)}$$

The number and composition of the clusters, and hence $H(L)$, changes for each cut of the cluster tree, but the entropy of the categories, $H(C)$, is constant. This function maximizes when these two entropies are most equal. Figure 3 shows the relationship between entropy and mutual information and the quantities defined for the gain ratio and revised ratio.

4.3 Percent Correct

Another method for determining the cut point is to count the number of items correctly classified for each cut of the tree. The number of correct items for a given cluster is equal to the number of items in the category with the maximum value for that cluster. For singleton clusters, an item is counted as correctly categorized if the the category is also a singleton. The percent correct is the total number of correct items divided by the total number of items. This value is useful for comparing results between cluster trees as well as for finding the cut point.

5 Unsupervised Method

An unsupervised method that worked well was to select a cut point that maximizes the number of clusters formed within a specified size range. Let s be an estimate of the size of the cluster and $r < 1$ the range. The algorithm counts the number of clusters at height T that fall within the interval:

$$[s * (1 - r), s * (1 + r)]$$

The optimum cut point is at the height that has the most clusters in this interval.

The value of $r = .8$ was constant for both right and left cluster trees. For right contexts, where we expected many small clusters, $s = 8$, giving a size range of 2 to 14 items in a cluster. For left contexts, where we expected a few large clusters, $s = 100$, giving a size range of 20 to 180. The expected cluster size is the only assumption we make to adjust the cut point given the disparity in the expected number of categories for left and right contexts. A fully unsupervised method would not make an assumption about expected size.

6 Results

While our goal was to use a supervised method to evaluate the performance of the unsupervised method, the supervised functions we tested differed widely in their ability to predict the optimum cut point for the left and right context trees. The performance of the gain ratio, revised ratio, and percent correct are compared to the unsupervised method on left and right context cluster trees in Figures 4 and 5. The x axis gives the height at which the cut is evaluated by the function, and the y axis is the scaled value of the function for that height. The optimum cut point indicated by each function is at the height for which the function has the maximum value. These heights are given in Table 3. For the right context tree, for which the optimum cut produces many small clusters, there is general agreement between the unsupervised and supervised methods. For the left context tree, for which the optimum cut produces a few large clusters, there is a lack of agreement among the supervised methods with the gain ratio failing to indicate a meaningful cut. The maximum for the unsupervised method falls between the maxima for the revised ratio and percent correct. Based on these results, we used the unsupervised maximum to select the cut point for the left and right context cluster trees.

There are four steps in the analysis of the data. First, the cutpoint for the left context

Table 3: Heights for maximum values

Supervised Methods	Left Contexts	Right Contexts
Gain ratio	1	12
Revised ratio	33-34	11
Percent correct	15	10
Unsupervised	18-22	12

tree is determined. Second, the right similarity matrix is enhanced with data from left context clusters. Third, the cut point for the enhanced right context tree is determined. Finally, the verbs are cross-classified by left and right context cluster membership.

Step 1: We select the cut point for the left context tree at height $T = 18$, the unsupervised maximum. This cut creates clusters that are 90.7% correct as compared to 91.5% at height $T = 15$. At $T = 18$ there are 20 clusters for 6 inflectional categories, of which 6 are in the size range 20-180 specified by the unsupervised method.

Step 2: Reasoning that two verbs in the same cluster for inflectional category should be in different clusters for lemmas, [7] we created a new similarity matrix for the right context by increasing the distance between each pair of verbs that occurred in the same left context cluster. The distance was increased by substituting a constant equal to the value of the maximum distance between verbs. The number of verbs correctly classified increased from 63.5% for the original right context tree to 74.7% for the enhanced right context tree.

Step3: We select the cut point for the enhanced right context tree at height $T = 12$, the unsupervised maximum. This cut creates clusters that are 72.2% correct as compared to 74.7% at height $T = 10$. There are 155 clusters at height $T = 12$, which is 21 less than the total number of lemmas in the data set. 29% of the clusters are singletons which is lower than the proportion of singleton lemmas (45%). 60% of the singleton clusters contain singleton lemmas and are counted as correct.

Step 4: Each left context cluster was given a unique left ID and labeled for the dominant inflectional category. Each right context cluster was given a right ID and labeled for the dominant lemma. By identifying the left and right cluster membership for each verb, we were able to predict the correct inflectional category and lemma for 67.5% of the 400 verbs. Table 4 shows a set of consecutive right clusters in which the lemma and inflectional category are correctly predicted for all but one verb.

[7] This is true except for a few verbs that have free variation between certain regular and irregular forms.

Table 4: Verb paradigms

Cluster ID	Label	Verb
Right 68	CALL	
Left 12	VBZ	calls/VBZ
Left 19	VBG	calling/VBG
Left 11	VBN	called/VBN
Left 8	VBD	called/VBD
Left 7	VB	call/VB
Right 69	ADD	
Left 19	VBG	adding/VBG
Left 6	VBZ	added/VBD
Left 7	VB	add/VB
Right 70	COME	
Left 13	VBG	coming/VBG
Left 10	VBZ	comes/VBZ
Left 1	VBP	come/VBP
Left 5	VBN	come/VBN
Left 7	VB	come/VB
Left 8	VBD	came/VBD
Right 71	MAKE	
Left 19	VBG	making/VBG
Left 10	VBZ	makes/VBZ
Left 1	VBP	make/VBP
Left 7	VB	make/VB
Left 8	VBD	made/VBD

6.1 Conclusion

We have clearly demonstrated that a surprising amount of information about the verb paradigm is strictly local to the verb. We believe that distributional information combined with morphological information will produce extremely accurate classifications for both regular and irregular verbs. The fact that information consisting of nothing more than bigrams can capture syntactic information about English has already been noted by (Brown et al. 1992). Our contribution has been to develop a largely unsupervised method for cutting a cluster tree that produces reliable classifications. We also developed an unsupervised method to enhance the classification of one cluster tree by using the classification of another cluster tree. The verb paradigm is extracted by cross classifying a verb by lemma and inflectional category. This method depends on the successful classification of verbs by lemma and inflectional category sep-

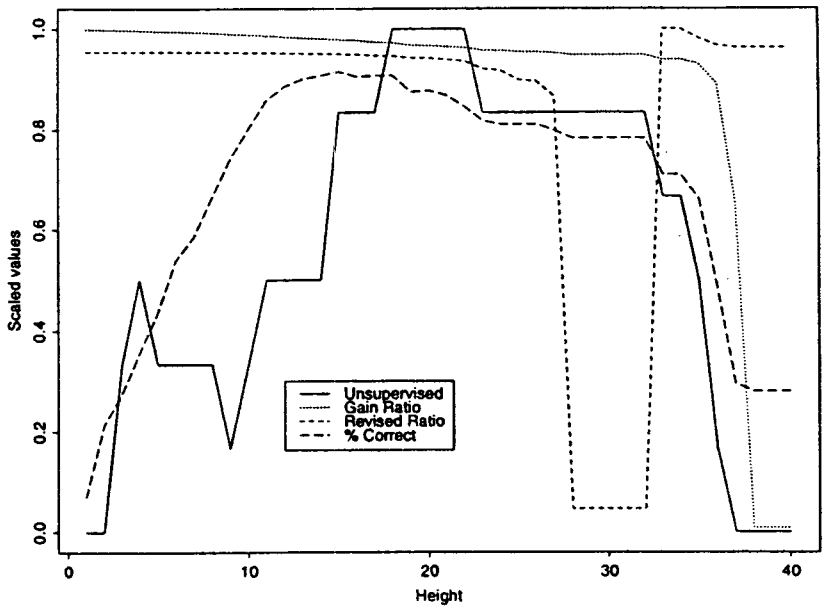

Figure 4: Value by height for left contexts

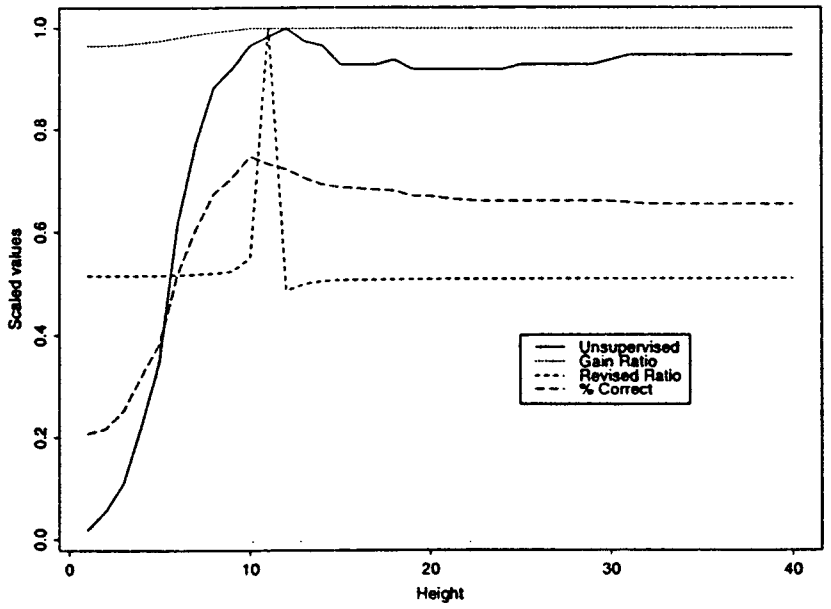

Figure 5: Value by height for right contexts

arately.

We are encouraged by these results to continue working towards fully unsupervised methods for extracting verb paradigms using distributional information. We hope to extend this exploration to other languages. We would also like to explore how the dual mechanisms of encoding verb lemmas and inflectional categories both by distributional criteria and by morphology can be exploited both by the child language learner and by automatic grammar extraction processes.

References

E. Brill, D. Magerman, M. Marcus, and B. Santorini. 1990. Deducing linguistic structure from the statistics of large corpora. *Proceedings of the DARPA Speech and Natural Language Workshop.* pp. 275-282.

E. Brill and S. Kapur. 1993. An information-theoretic solution to parameter setting. *Proceedings of the Georgetown University Round Table on Languages and Linguistics: Session on Corpus Linguistics.*

P.F. Brown, P.V. deSouza, R.L. Mercer, V.J. Della Pietra, J.C. Lai. 1992. Class-based n-gram models of natural language. *Computational Linguistics* 18:467-480.

T. Cover and J. Thomas. 1991. *Elements of Information Theory.* Wiley: New York.

I. Dagan, F. Pereira, and L. Lee. 1994. Similarity-based estimation of word cooccurrence probabilities. *Proceedings of the 32nd Annual Meeting of the ACL.* Las Cruces, NM. pp. 272-278.

S. Finch and N. Chater. 1992. Bootstrapping Syntactic Categories. *Proceedings of the 14th Annual Conference of the Cognitive Science Society of America.* Bloomington, Indiana. pp. 820-825.

F. Jelinek. 1990. Self-organized language modeling for speech recognition. In Waibel & Lee (Eds.) *Readings in Speech Recognition* Morgan Kaufmann Pub., SanMateo, CA.

D.M. Magerman and M.P. Marcus. 1990. Parsing a natural language using mutual information statistics. *Proceedings of AAAI-90.* Boston, MA.

F. Pereira, N. Tishby, and L. Lee. 1993. Distributional clustering of English words, In *Proceedings of the 30th Annual Meeting of the ACL.* Columbus, OH. pp. 183-190.

S. Pinker. 1984. *Language Learnability and Language Development.* Harvard University Press, Cambridge, MA.

J. R. Quinlan. 1990. Induction of Decision Trees. In J. W. Shavlik and T.G. Dietterich (Eds.) *Readings in Machine Learning.* Morgan Kaufmann, Pub., SanMateo, CA.

A. Ratnaparkhi. 1997. *A simple introduction to maximum entropy models for natural language processing.* Technical Report 97-08, Institute for Research in Cognitive Science, University of Pennsylvania.

S. Russell and P. Norvig. 1995. *Introduction to Artificial Intelligence.* Prentice Hall, Englewood Cliffs, NJ. pp. 540-541

G.A.F. Seber. 1984. *Multivariate Observations.* John Wiley & Sons: New York. pp. 347-391.

H. Schuetze. 1993. Part-of-speech induction from scratch. *Proceedings of the 31st Annual Meeting of the ACL.* Columbus, OH.