A NOVEL WORD CLUSTERING ALGORITHM BASED ON LATENT SEMANTIC ANALYSIS

Jerome R. Bellegarda, John W. Butzberger, Yen–Lu Chow, Noah B. Coccaro, and Devang Naik*

Interactive Media Group
Apple Computer, Inc.
Cupertino, California 95014

ABSTRACT

A new approach is proposed for the clustering of words in a given vocabulary. The method is based on a paradigm first formulated in the context of information retrieval, called *latent semantic analysis*. This paradigm leads to a parsimonious vector representation of each word in a suitable vector space, where familiar clustering techniques can be applied. The distance measure selected in this space arises naturally from the problem formulation. Preliminary experiments indicate that the clusters produced are intuitively satisfactory. Because these clusters are semantic in nature, this approach may prove useful as a complement to conventional class-based statistical language modeling techniques.

I. INTRODUCTION

Stochastic language modeling plays a central role in large vocabulary speech recognition. In a typical application, the purpose of an n-gram language model may be to constrain the acoustic analysis, guide the search through various (partial) text hypotheses, and/or contribute to the determination of the final transcription. Success in these endeavors depends on the ability of the language model to suitably discriminate between different strings of n words. This ability is in turn critically influenced by the two familiar issues of coverage and estimation.

The coverage issue reflects the fact that current systems cannot recognize any "unknown" word. The vocabulary must therefore be chosen so that the expected text (e.g., to be dictated) has as few unknown words as possible [1]. In this paper we primarily address the other issue, which arises from practical constraints on the size of available text databases. Many events (i.e., occurrences of n-word strings) are seen infrequently, yielding questionable probabilities; hence the need for fairly sophisticated parameter estimation and smoothing [2]. One common solution is to group words into classes and accumulate statistics at the class level rather than the word level. This makes the frequency counts more reliable and thereby improves the robustness of the estimation (e.g., see [3]). Broadly speaking, the underlying strategy is to better estimate the conditional probability of a word given some context by taking advantage of observations of other words that behave "like" this word in this particular context.

A number of variants have been developed on this theme, using grammatical constraints such as part-of-speech, or morphological units such as lemma, or both [4]. More recently, algorithms have evolved to automatically determine word classes without explicit syntactic or semantic knowledge: cf, e.g., [5], [6]. In [5], for example, all words are gathered into a single class at the beginning of the procedure, and are successively split to maximize the average mutual information of adjacent classes. In [6], a similar divisive clustering is proposed, based on binomial posteriori distributions on word co-occurrences. A number of other authors have described related approaches, with different variations in the optimization criterion or distance metric used for clustering.

This paper proposes an alternative framework to word clustering based on a paradigm originally formulated in the context of information retrieval, called latent semantic analysis [7]. The paper is organized as follows. In the next section we discuss our general strategy to perform word clustering using this alternative framework. In Section III, we present the vector representation derived from latent semantic analysis. Section IV describes the distance measure used to implement the clustering. Finally, in Section V we report on preliminary experimental results and give some examples of the clusters obtained.

II. GENERAL STRATEGY

The basic premise of the latent semantic framework

[1] N.B. Coccaro is with the Dept. of Computer Science, University of Colorado at Boulder, Boulder, CO 80309.

is that co-occurrence analysis should take place across larger contexts (longer spans) than traditionally considered (i.e., a bigram as in [3] or a trigram as in [6]). The span of choice is a *document*, which can be defined as a semantically homogeneous set of sentences embodying a given storyline. This amounts to a much looser definition of co-occurrence, where two words are said to co-occur if they tend to appear in the same kind of documents, whether or not they actually occur within identical word contexts in those documents. An important benefit of this generalization is the natural integration of long-term dependencies into the clustering procedure. By the same token, this approach is better suited at discovering semantic than syntactic clusters. It should therefore be viewed as complementing, rather than replacing, some of the methods mentioned above.

To take advantage of the concept of document, we of course have to assume that the available training data is tagged at the document level, i.e., there is a way to identify article boundaries. This is the case, for example, with the ARPA North American Business (NAB) News corpus [8]. This assumption enables the construction of a matrix of co-occurences between words and documents (articles). This matrix is accumulated from the available training data by simply keeping track of which word appeared in what document. Note that, in marked contrast with n-gram modeling, this step ignores word order. This is because latent semantic analysis seeks to relate to one another those words which are found to be semantically linked from the evidence presented in the training text database, without regard to the particular syntax used to express that semantic link.

After the word-document matrix of co-occurences is constructed, the algorithm proceeds in three steps. First, a singular value decomposition (SVD) is performed on this matrix, thus producing singular vector representations for words and documents. Second, a distance measure consistent with the SVD formalism is defined on the resulting vector space. Third, the singular vectors are clustered according to this metric.

The role of the SVD step is to establish a one-to-one mapping between a word in the given vocabulary and a vector in some space of appropriate dimension. Specifically, this space is spanned by the (left) singular vectors resulting from the SVD. An important property of this space is that two words whose representations are "close" (in some suitable metric) tend to appear in the same kind of documents, whether or not they actually occur within identical word contexts in those documents. This establishes a systematic framework where familiar clustering techniques can be applied. The end result is a set of word clusters, each of which can be assumed to represent a reasonably homogeneous semantic event.

III. VECTOR REPRESENTATION

Let $\mathcal{V}$, $|\mathcal{V}| = M$, be some vocabulary of interest and $\mathcal{T}$ a training text corpus, i.e., a collection of N articles (documents) from a variety of sources. (Typically, M and N are on the order of ten and hundred thousand, respectively; $\mathcal{T}$ might comprise a couple hundred million words.) The task at hand is to define a mapping between the set $\mathcal{V}$ and a vector space $\mathcal{S}$, whereby each word in $\mathcal{V}$ is represented by a vector in $\mathcal{S}$.

We first construct a word-document matrix K associated with $\mathcal{V}$ and $\mathcal{T}$. This is done by computing, for each word $w_i \in \mathcal{V}$, the weighted count K_{ij} of w_i in each of the documents $d_j \in \mathcal{T}$. Following results from information retrieval (cf., e.g., [9]), this weighted count is expressed as:

$$K_{ij} = G_i \, L_{ij} \ , \tag{1}$$

where G_i is a global weight, indicating the overall importance of w_i as an indexing term for the collection $\mathcal{T}$, and L_{ij} is a local value, which may reflect a possible normalization within d_j.

The global weighting G_i translates the fact that two words appearing with the same count in d_j do not necessarily convey the same amount of information about the document; this is subordinated to the distribution of the words in the collection $\mathcal{T}$. Let us denote by c_{ij} the number of times w_i occurs in document d_j, and by t_i the total number of times w_i occurs in the entire collection $\mathcal{T}$. Then the relative frequency of w_i in d_j is obtained as:

$$f_{ij} = \frac{c_{ij}}{t_i} \ , \tag{2}$$

and the associated normalized entropy of w_i is seen to be:

$$E_i = - \frac{1}{\log(N)} \sum_{j=1}^{N} f_{ij} \log f_{ij} \ . \tag{3}$$

By definition, $0 \leq E_i \leq 1$, with equality if and only if $f_{ij} = 1$ and $f_{ij} = 1/N$, respectively. A value of E_i close to 1 underscores a word distributed across many documents throughout the corpus, and therefore of little indexing value. Conversely, a value of E_i close to 0 indicates a word present only in a few specific documents, i.e., of suitable indexing value. Hence, $G_i = 1 - E_i$ is a reasonable global weight for the word w_i.

The local value L_{ij} is a transformed version of c_{ij} which may reflect any adjustment to the raw count c_{ij}. For example, it is common to use $L_{ij} = \log(1 + c_{ij})$, where the log dampens the effects of large differences in counts [9]. It is also possible to normalize for document length. If we denote by n_j the number of words in document d_j, then:

$$L_{ij} = \log_2\big(1 + \frac{c_{ij}}{n_j}\big) \ . \tag{4}$$

is such that $0 \leq L_{ij} \leq 1$. In practice, $L_{ij} = c_{ij}$ seems to work essentially as well as any other functional.

The $(M \times N)$ word-document matrix K with entries K_{ij} given by (1) fully describes, for the training corpus $\mathcal{T}$, which words appeared in what contexts. We then proceed to perform a singular value decomposition of K as follows:

$$K \approx \hat{K} = U\, S\, V^T\,, \tag{5}$$

where U is the $(M \times R)$ matrix of left singular vectors u_i $(1 \leq i \leq M)$, S is the $(R \times R)$ diagonal matrix of singular values, V is the $(N \times R)$ matrix of right singular vectors v_j $(1 \leq j \leq N)$, $R \ll M (\ll N)$ is the order of the decomposition, and ^{T} denotes matrix transposition. The ith left singular vector u_i can be viewed as the representation of w_i in a vector space of dimension R. This is the space $\mathcal{S}$ which we sought. Note that the jth right singular vector v_j can be viewed as the representation of d_j in the *same* vector space $(\mathcal{S})$ of dimension R. This dimension is bounded from above by the rank of the matrix K, and from below by the amount of distortion tolerable in the decomposition. Reasonable values of R are 100 to 200.

The basic idea behind (5) is that $\hat{K}$ captures the major associational structure in K and ignores higher order effects. As a result, the "closeness" of vectors in $\mathcal{S}$ is determined by the overall pattern of the language used in $\mathcal{T}$, as opposed to specific constructs. In particular, this means that two words which do not co-occur in $\mathcal{T}$ will still be "close" if that is otherwise consistent with the major patterns of the language (e.g., if they tend to co-occur with a common set of words). This has the important benefit of alleviating the effects of polysemy.

IV. CLUSTERING

In the vector space $\mathcal{S}$ obtained above, each word $w_i \in \mathcal{V}$ is represented by the associated left singular vector of dimension R, u_i. To be able to cluster such vectors, we first need to define a distance measure on $\mathcal{S}$. Since the extent to which words have a similar pattern of occurrence across the entire set of documents can be inferred from:

$$\hat{K}\,\hat{K}^T = U\, S^2\, U^T\,, \tag{6}$$

a natural metric to consider for the distance between u_i and u_j is the cosine of the angle between $u_i S$ and $u_j S$. Thus:

$$D(u_i, u_j) = \cos(u_i S, u_j S) = \frac{u_i\, S^2\, u_j^T}{\|u_i S\|\, \|u_j S\|}\,, \tag{7}$$

for any $1 \leq i, j \leq M$.

Once this metric is specified, it is straightforward to proceed with the clustering of the vectors u_i, using any of a variety of algorithms; see, e.g., [10]. Since the number of such vectors is relatively large, it is advisable to perform this clustering in stages, using, for example, K-means and bottom-up clustering sequentially. In that case, K-means clustering is used to obtain a coarse partition of the vocabulary $\mathcal{V}$ in to a small set of superclusters. Each supercluster is then itself partitioned using bottom-up clustering. The result of this process is a set of clusters C_k, $1 \leq k \leq L$, each of which can be characterized by an appropriate multivariate distribution in the space $\mathcal{S}$.

The quality of this clustering can be objectively measured by the perplexity of the associated language model on some test text. In the present case, if we denote by Q the total number of words in the text, the latter is given by:

$$PP = \exp\Big(-\frac{1}{Q}\sum_{i=1}^{Q} \log \Pr(w_i|\mathcal{C}, \mathcal{S})\Big)\,, \tag{8}$$

where $\mathcal{C}$ is the context considered, and $\mathcal{S}$ indicates the particular vector space selected in the word representation. The probability can then be expanded as:

$$\Pr(w_i|\mathcal{C}, \mathcal{S}) = \Pr(w_i|C_i)\, \Pr(C_i|\mathcal{C})\,, \tag{9}$$

where C_i is the cluster in $\mathcal{S}$ which w_i belongs to. In this expression, $\Pr(w_i|C_i)$ can be obtained through the multivariate distribution mentioned in the preceding paragraph, and $\Pr(C_i|\mathcal{C})$ can be obtained from the relative frequency of cluster C_i in the context $\mathcal{C}$. Note that this latter quantity may vary significantly depending on the choice of the context considered, which makes perplexity measurements somewhat hard to interpret. For the sake of brevity we will omit any further details on this matter.

V. EXPERIMENTAL RESULTS

We have performed preliminary experiments to illustrate some of the potential benefits of the above approach. We considered a subset $\mathcal{T}$ of the NAB News corpus [8], composed of about $N = 17,500$ documents, comprising approximately 10 million words. These articles were selected randomly from the Wall Street Journal portion of the corpus, in such a way that all years from 1987 to 1993 are represented. (In addition, about 2 million words from 1992 and 1994, selected in a similar fashion, were set aside for test purposes.) The vocabulary $\mathcal{V}$ was constructed by taking the 20,000 most frequent words of the NAB News corpus, and removing approximately 300 non-content words (stop words), for a total of $M = 19,700$ words.

This led to a $(19,700 \times 17,500)$ word-document matrix of co-occurences, stored in sparse fashion. We performed the singular value decomposition of this matrix

using the single vector Lanczos method implemented by Berry [11]. The number of singular values retained (i.e., the dimension of the associated vector space) was set to 125, which seemed to achieve an adequate balance between reconstruction error and noise suppression. We clustered the vectors in this space into 100 superclusters of approximately 200 vectors each using simple K-means clustering. We then refined each of the superclusters into 20 clusters each using bottom-up clustering. This produced a set of 2000 clusters, each comprising about 10 words on average. Finally, we merged related clusters from different superclusters back together to avoid excessive fragmentation. This resulted in a cluster set of size 500.

To show what these clusters look like, we selected two examples of the clusters so obtained.

• **Cluster 1**: *abstract, art, artist, artist's, canvas, curator, decorative, drawings, exhibit, exhibition, exhibitions, gallery, galleries, Gogh, Henri, museum, museum's, museums, painted, painter, painters, painting, paintings, photographs, Picasso, poems, Pollock, Pons, portraits, retrospective, Revere, sketches*

• **Cluster 2**: *appeal, appeals, appellate, argued, arguments, attorney's, circuit, confessed, count, courts, criminal, decide, decision, indict, indictments, judge, judge's, judges, leniency, misdemeanor, office's, prosecuted, prosecution, prosecutions, overturned, prosecutor, prosecutorial, prosecutors, ruled, ruling, rulings, witness*

The first thing to note is that these clusters comprise words with different part of speech, a marked difference with conventional class n-gram techniques (cf. [3]–[6]). This is a direct consequence of the semantic nature of the derivation. Second, some obvious words seem to be missing from the clusters: for example, the singular noun "drawing" from cluster 1 and the present tense verb "rule" from cluster 2. This is one of the effects of polysemy: "drawing" and "rule" are more likely to appear in the training text with their alternative meanings (as in "drawing a conclusion" and "breaking a rule," respectively), thus resulting in different cluster assignments. Finally, some words seem to contribute only marginally to the clusters: for example, "poems" from cluster 1 and "office's" from cluster 2. These are the unavoidable outliers at the periphery of the clusters.

VI. CONCLUSION

We have described a word clustering approach based on the latent semantic analysis paradigm first formulated in information retrieval. One of the advantages of this framework is that it results in a vector representation of each word in a space of relatively modest dimension. This allows well-known clustering algorithms to be applied efficiently. As the name implies, the proposed approach results in clusters that are semantic in nature, by exploiting large span relationships between words. In contrast, conventional class-based modeling techniques rely on short span relationships, and therefore tend to produce syntactically-oriented clusters. Thus, the two paradigms have the potential of complementing each other. We are currently exploring some of these interactions.

REFERENCES

[1] F. Jelinek, "Self–Organized Language Modeling for Speech Recognition," *Readings in Speech Recognition*, A. Waibel and K.F. Lee, Eds, Morgan Kaufmann Publishers, pp. 450–506, 1990.

[2] S.M. Katz, "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer," *IEEE Trans. Acoust. Speech Signal Proc.*, Vol. ASSP–35, pp. 400–401, March 1987.

[3] U. Essen and V. Steinbiss, "Co–Occurrence Smoothing for Stochastic Language Modeling," in *Proc.* 1992 *ICASSP*, San Francisco, CA, pp. 161–164, March 1992.

[4] G. Maltese and F. Mancini, "An Automatic Technique to Include Grammatical and Morphological Information in a Trigram–Based Statistical Language Model," in *Proc.* 1992 *ICASSP*, San Francisco, CA, pp. 157–160, March 1992.

[5] M. Jardino and G. Adda, "Automatic Word Classification Using Simulated Annealing," in *Proc.* 1993 *ICASSP*, Minneapolis, MN, pp. 41–44, May 1993.

[6] M. Tamoto and T. Kawabata, "Clustering Word Category Based on Binomial Posteriori Co–Occurrence Distribution,"in *Proc.* 1995 *ICASSP*, Detroit, MI, pp. 165–168, May 1995.

[7] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," *J. Am. Soc. Inform. Science*, Vol. 41, pp. 391–407, 1990.

[8] F. Kubala *et al.*, "The Hub and Spoke Paradigm for CSR Evaluation", in *Proc.* ARPA *Speech and Natural Language Workshop*, Morgan Kaufmann Publishers, pp. 40–44, March 1994.

[9] S.T. Dumais, "Improving the Retrieval of Information from External Sources," *Behavior Res. Methods, Instrum., Computers*, Vol. 23, No. 2, pp. 229–236, 1991.

[10] J.R. Bellegarda, "Context–Dependent Vector Clustering for Speech Recognition," Chapter 6 in *Speech and Speaker Recognition: Advanced Topics*, C.-H. Lee, F.K. Soong, and K.K. Paliwal, Eds, New York: Kluwer Academic Publishers, March 1996.

[11] M.W. Berry, "Large–Scale Sparse Singular Value Computations," *Int. J. Supercomp. Appl.*, Vol. 6, No. 1, pp. 13–49, 1992.