A Bernoulli mixture model for word categorisation *

J. González, A. Juan, P. Dupont[§], E. Vidal and F. Casacuberta

Dep. de Sist. Informáticos y Computación,
Instituto Tecnológico de Informática,
Universidad Politécnica de Valencia,
Camino de Vera s/n, 46071 Valencia
{jgonza,ajuan,evidal,fcn}@iti.upv.es

[§]EURISE
Université Jean Monnet
23, rue P. Michelon
42023 Saint-Etienne Cedex - France
pdupont@univ-st-etienne.fr

Abstract

The problem of word categorisation is formulated as one of unsupervised mixture modelling where Bernoulli distributions capture contextual information. We detail how the free parameters of the mixture models can be estimated through an *EM* procedure. A deterministic word-to-class mapping is derived from this model using a hierarchical clustering algorithm.

Keywords: Word categorisation, Bernoulli mixture model, clustering, language modelling.

1 Introduction

Categorisation plays an important role in language modelling for natural language processing [1], continuous speech recognition [2], continuous handwritten character recognition [1], topic identification [3] and machine translation [4]. By categorisation we mean the substitution of words or group of words by labels (*word-classes*) that can represent their syntactic or semantic class within a limited rank of options.

On the one hand, in the learning of language models from a corpus, categorisation let us reduce the number of free parameters to be estimated. On the other hand, the use of word-classes may allow us to easily increase the vocabulary of the task, without requiring to train the whole system again.

One problem with categorisation is the selection of an adequate set of word-classes. This set can be *a priori* defined from (linguistic) knowledge of the considered task. But, in the probabilistic framework, some algorithms have been proposed to estimate the most adequate set of word-classes from a training corpus [2]. These algorithms can also be used in combination with grammar induction techniques [5, 6].

In this paper, we introduce a new method for seeking word-classes from a training set. The main idea is to represent each word with some contextual information modelled by a mixture of Bernoulli distributions. A deterministic word-to-class mapping is derived from this model using a hierarchical clustering algorithm [7]. Some preliminary experiments illustrate the interest of the proposed method.

*Work supported by the Spanish MCT under grants TIC2000-1703-CO3-01 and TIC 1FD1997-1433

2 Proposed approach

2.1 Probabilistic framework

The problem of word categorisation is formulated as one of unsupervised mixture modelling. On the one hand, we assume that category membership of a word in an input sequence only depends on the identity of its immediate surrounding words or *context*, but not on their respective positions. Thus, if $W = \{w_1, \ldots, w_d\}$ denotes the lexicon from which words are extracted, then the context of a word in an input sequence can be represented as an indicator vector $\boldsymbol{x} \in \{0,1\}^d$, such that $x_i = 1$ if w_i belongs to the context, $x_i = 0$ otherwise.

On the other hand, it is also assumed that (word) contexts come from a mixture of a known number c of category-components,

$$P(\boldsymbol{x} \mid \boldsymbol{\Theta}) = \sum_{j=1}^{c} \pi_j \, P_j(\boldsymbol{x} \mid \boldsymbol{\Theta}_j), \tag{1}$$

where each P_j is a multivariate Bernoulli distribution parameterized by $\boldsymbol{\Theta}_j$,

$$P_j(\boldsymbol{x} \mid \boldsymbol{\Theta}_j) = \prod_{k=1}^{d} \Theta_{jk}^{x_k} \, (1 - \Theta_{jk})^{1-x_k}. \tag{2}$$

Each distribution P_j identifies a certain context type where the kth component of $\boldsymbol{\Theta}_j$ represents the probability of word w_k to be present. The unknown parameters are $\boldsymbol{\Theta} = (\pi_1, \ldots, \pi_c, \boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_c)$, that is, the mixing coefficients, $\pi_1, \ldots, \pi_c$, with $\sum_{j=1}^{c} \pi_j = 1$, and the parameters of the c Bernoulli distributions, $\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_c$, which are assumed to be independent.

2.2 Context distribution modelling

From the assumptions described in section 2.1, estimating a (word) context distribution model consists in learning the unknown parameters from a given training dataset, $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$. Following the maximum likelihood principle, the best parameter values maximize the log-likelihood function of $\boldsymbol{\Theta}$,

$$\mathcal{L}(\boldsymbol{\Theta} \mid X) = \sum_{i=1}^{n} \log \left(\sum_{j=1}^{c} \pi_j \, P_j(\boldsymbol{x}_i \mid \boldsymbol{\Theta}_j) \right). \tag{3}$$

In order to find these optimal values, it is useful to think of each context $\boldsymbol{x}_i$ as an *incomplete* categorised context, which can be completed by an indicator vector

$\boldsymbol{z}_i = (z_{i1}, \ldots, z_{ic})^t$ with 1 in the position corresponding to the component generating $\boldsymbol{x}_i$ and zeros elsewhere. In doing so, a complete version of the log-likelihood function (3) can be stated as

$$\mathcal{L}_C(\boldsymbol{\Theta} \mid X, Z) = \sum_{i=1}^{n} \sum_{j=1}^{c} z_{ij} \, (\log \pi_j + \log P_j(\boldsymbol{x}_i \mid \boldsymbol{\Theta}_j)) \,, \tag{4}$$

where $Z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n\}$ is the so-called *missing* data.

The form of the log-likelihood function given in (4) is generally preferred because it makes available the well-known *EM* optimization algorithm (for finite mixtures) [8]. This algorithm proceeds iteratively in two steps. The E(xpectation) step computes the expected value of the missing data given the incomplete data and the current parameters. The M(aximization) step finds the parameter values which maximize (4), on the basis of the missing data estimated in the E step. In our case, the E step replaces each z_{ij} by its expected value,

$$z_{ij} = \frac{\pi_j \, P_j(\boldsymbol{x}_i \mid \boldsymbol{\Theta}_j)}{\sum_{k=1}^{c} \pi_k \, P_k(\boldsymbol{x}_i \mid \boldsymbol{\Theta}_k)} \qquad (i = 1, \ldots, n; \;\; j = 1, \ldots, c), \tag{5}$$

while the M step finds the maximum likelihood estimates for the mixing coefficients,

$$\pi_j = \frac{\sum_{i=1}^{n} z_{ij}}{n} \qquad (j = 1, \ldots, c), \tag{6}$$

and the parameters of the c Bernoulli distributions,

$$\boldsymbol{\Theta}_j = \frac{\sum_{i=1}^{n} z_{ij} \, \boldsymbol{x}_i}{\sum_{i=1}^{n} z_{ij}} \qquad (j = 1, \ldots, c). \tag{7}$$

An initial value for the missing data or the parameters is required for the EM algorithm. To do this, we simply pick c "seed" contexts at random and fill in the missing data by assigning each context to its nearest (Hamming distance) seed. From this point on, each iteration of the EM algorithm is guaranteed not to decrease the log-likelihood function and the algorithm is guaranteed to converge to a local maximum.

2.3 Probabilistic word categorisation

Let $\hat{\boldsymbol{\Theta}}$ be the maximum likelihood estimate for $\boldsymbol{\Theta}$ found by the EM algorithm. Given a (word) context $\boldsymbol{y}$, our probabilistic categorisation model assigns it to the jth category with probability

$$\hat{z}_j(\boldsymbol{y}) = \frac{\pi_j \, P_j(\boldsymbol{y} \mid \hat{\boldsymbol{\Theta}}_j)}{\sum_{k=1}^{c} \pi_k \, P_k(\boldsymbol{y} \mid \hat{\boldsymbol{\Theta}}_k)} \,. \tag{8}$$

A possible drawback of this model is that it ignores the identity of the words originating their corresponding contexts. So, very similar categorisation rules can be obtained for contexts associated with different words but, on the contrary, distinct rules can be obtained for contexts arising from the same word. On the other hand, we can force contexts of the same word to have similar categorisation rules by simply specializing each category component to a single word (supervised learning). However, this would imply having at least one category per word, thus reducing model usefulness. Therefore, the former (unsupervised) approach seems to be preferable to the latter alternative.

2.4 An approximate deterministic categorisation rule

Although model (8) establishes a well-founded categorisation rule, a deterministic mapping may be useful in some cases [6]. To do this, a single (probabilistic) categorisation rule is first obtained for each word w_k, $k = 1, \ldots, d$, by simply averaging (8) over all training contexts coming from w_k. More precisely, if X_k denotes the set of such contexts, then w_k will be assigned to the jth category with probability

$$Pr(w_k \in j\text{th category}) = \frac{1}{\mid X_k \mid} \sum_{\boldsymbol{x} \in X_k} \hat{z}_j(\boldsymbol{x}) \tag{9}$$

Note that equation (9) represents each word in terms of a probability function over the c possible categories. Doing so, words used in similar contexts would have similar representations, while words used in different contexts would have different representations too. From this point of view, it is straightforward to compute a deterministic categorisation rule into, say, C categories, by choosing an appropriate distance function to compare word representations and a standard distance-based clustering technique. For instance, the experiments reported in the next section are based on the so-called *Hellinger's distance* and *Ward's agglomerative hierarchical clustering* method [7].

3 Experiments

In order to set a preliminary idea of the goodness of our approach, we applied it to the MLA task, which is rather simple. The task corpus consists of 1000 English sentences which describe and manipulate visual scenes, and the associated vocabulary is composed of 25 words. Let us see a series of examples of this kind of sentences:

- a medium dark circle is added far to the right of the large triangle and the large dark square

- the large circle which is far to the left of the dark circle and the medium light square is removed
- a dark triangle is added far above the medium dark triangle

Experiments were carried out for different sizes of context window, and varying the number c of category-components. The qualitatively best results were achieved taking into account two words at both sides of the current word and a mixture of 16 Bernoulli distributions. The associated dendrogram can be seen in Fig. 1.

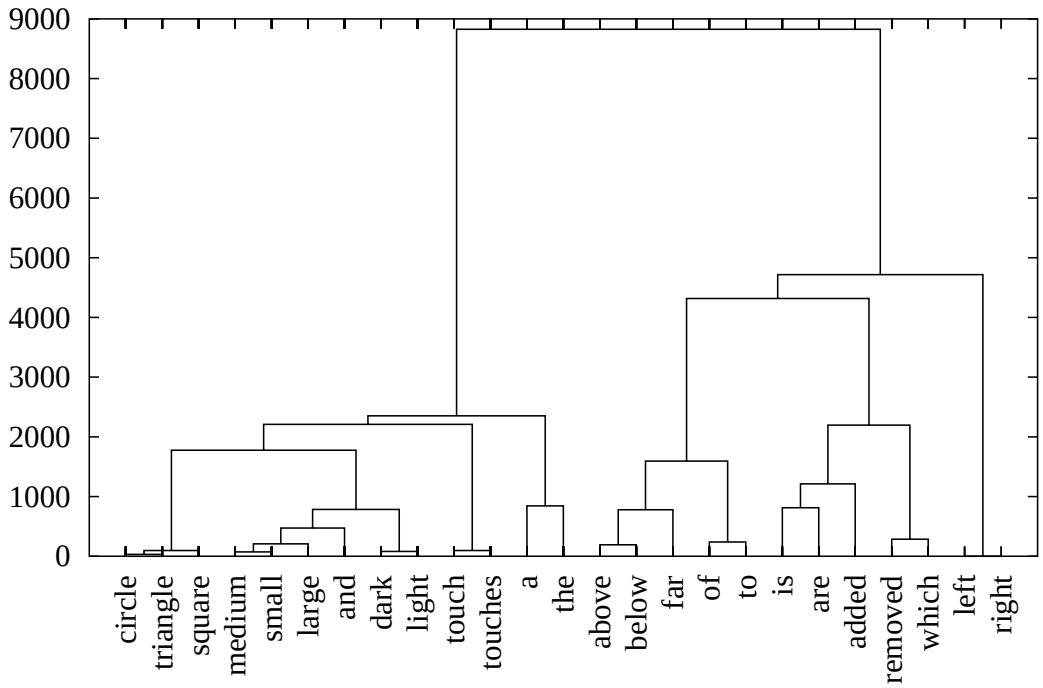

Figure 1: Best dendrogram obtained for the MLA task.

Several word clusters are meaningful like, for instance, `circle-triangle-square` (geometrical figures), `medium-small-large` (types of size), `dark-light` (adjectives of brightness), `touch-touches` (from the verb *to touch*), `a-the` (articles), `above-below` (position), `of-to` (prepositions), `is-are` (from the verb *to be*) and `left-right` (direction).

Besides, the verb *to be* tends to be clustered with participles (`added` and `removed`) and the word `which` is also closely related as *which is/are* phrases are common in this

task. The position of *far* near the group `above-below` and the word `to` is interesting since the sequence *far [above/below] to* is also common. It is worth noting that, except for the words derived from *to touch*, all words related to noun phrases are placed on the left branch of the dendrogram.

4 Conclusions

The main problem of categorisation is the selection of an adequate set of word-classes. In this paper, we tackle this problem by means of an unsupervised method which uses contextual information modelled by a mixture of Bernoulli distributions. The proposed method was able to discover relevant word-classes in some preliminary experiments.

References

[1] T. Niessler. *Category-based statistical language models.* Ph.D. Thesis. University of Cambridge, 1997.

[2] S. C. Martin, J. Liermann, H. Ney. *Algorithms for Bigram and Trigram Word Clustering.* Proc. Europ. Conf. on Speech Communication and Technology, Madrid, Spain, pp. 1253-1256, 1995.

[3] G. Moore, S. Young. *Class-based language model adaptation using mixtures of word-class weight.* Proc. of ICSLP, 2000.

[4] Instituto Tecnológico de Informática. Tech. Report Deliverable D2.1a, EuTrans Example-Based Language Translation Systems, Information Technology, Long term Research Domain, Open Scheme, Project Number 30268, 2000.

[5] E. Vidal, F. Casacuberta and P. García. *Grammatical Inference and Speech Recognition.* New Advances and Trends in Speech Recognition and Coding, NATO ASI Series, Springer-Verlag, pp. 174-191, 1995.

[6] P. Dupont, L. Chase. *Using symbol clustering to improve probabilistic automaton inference.* Lecture Notes in Artificial Intelligence, No. 1433, Springer Verlag, Grammatical Inference, ICGI'98, pp. 232 – 243, 1998.

[7] R. O. Duda, P. Hart. *Pattern classification and scene analysis.* John Wiley, 1973.

[8] A. P. Dempster, N. M. Laird and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society B*, 39: 1–38, 1977.