Learning Classification with both Labeled and Unlabeled Data

Jean-Noël Vittaut, Massih-Reza Amini, Patrick Gallinari

Computer Science Laboratory of Paris 6 (LIP6),
University of Pierre et Marie Curie,
8 rue du capitaine Scott,
75015 Paris, France

{vittaut, amini, gallinari}@poleia.lip6.fr

Abstract. A key difficulty for applying machine learning classification algorithms for many applications is that they require a lot of hand-labeled examples. Labeling large amount of data is a costly process which in many cases is prohibitive. In this paper we show how the use of a small number of labeled data together with a large number of unlabeled data can create high-accuracy classifiers. Our approach does not rely on any parametric assumptions about the data as it is usually the case with generative methods widely used in semi-supervised learning. We propose new discriminant algorithms handling both labeled and unlabeled data for training classification models and we analyze their performances on different information access problems ranging from text span classification for text summarization to e-mail spam detection and text classification.

1 Introduction

Semi-supervised learning has been introduced for training classifiers when large amounts of unlabeled data are available together with a much smaller amount of labeled data. It has recently been a subject of growing interest in the machine learning community. This paradigm particularly applies when large sets of data are produced continuously and when hand-labeling is unfeasible or particularly costly. This is the case for example for many of the semantic resources accessible via the web. In this paper, we will be particularly concerned with the application of semi-supervised learning techniques to the classification of textual data. Document and text-span classification has become one of the key techniques for handling and organizing text data. It is used to find relevant information on the web, to filter e-mails, to classify news stories, to extract relevant sentences, etc. Machine Learning techniques have been widely used for classifying textual data. Most algorithms rely on the supervised learning paradigm and require the labeling of very large amounts of documents or text-spans which is unrealistic for large corpora and for on-line learning.

We introduce here a new semi-supervised approach for classification. The originality of our work lies in the design of a discriminant approach to semi-supervised

learning whereas others mainly rely on generative classifiers. Compared to the latter approach, our method does not rely on any parametric assumptions about the data, and it allows for better performance than generative methods especially when there are few training data. It finally leads to very simple and fast implementation. This approach is generic and can be used with any type of data, however, we focus here in our experiments on textual data which is of particular interest to us.

In our previous work, we were interested on the classification of text spans and more particularly sentences for text summarization using semi-supervised algorithms [1, 2, 3, 4].

In [1, 2], we have shown the link between CEM and the mean-squared error classification for simple linear classifiers and a sequential representation of documents. In [1] we gave a semi-supervised version of the algorithm and in [2] we extended the idea for unsupervised learning techniques.

In [3] we adopted a vectorial representation of sentences rather than a sequential representation and presented a discriminant semi-supervised algorithm in a more general setting of logistic classifiers and considered a binary decision problem. In [4] we studied, in detail, the application of this method to text summarization seen as a sentence extraction task and analyzed its performances on two real world data sets.

In this work, we extend this idea for any discriminant classifiers and for multi-class problems, and give results on more various information retrieval classification tasks such as e-mail filtering and web page classification.

The paper is organized as follows; we first make a brief review of work on text classification, ranking and filtering, and on recent work in semi-supervised learning (section 2). In section 3, we describe our semi-supervised approach for classification and present the formal framework of the model. Finally we present a series of experiments on the UCI e-mail spam collection, on a collection from NIPS-2001 workshop for text classification and on TIPSTER SUMMAC collection for text summarization by sentence extraction. For the latter, text summarization can be considered as an instance of sentence classification where text spans are labeled relevant or irrelevant for the summary [4].

2 Related work

2.1 Text classification and summarization

The field of text classification has been and is still very active. One of the early application of text classification was for author identification. The seminal work by [26] examined authorship of the different Federalist papers. More recently text classification has been applied to a wide variety of practical problems ranging from cataloging news articles to classifying web pages and book recommendation.

An early and popular machine learning technique for text classification is the naive Bayes approach [20]. But a variety of other machine learning techniques has been applied to text classification. Recently support vector machines have attracted much

work for this task [15, 17]. Other authors have used neural networks [34] and a variety of boosting approaches [29]. But until now, no single technique has emerged as clearly better than the others, though some recent evidence suggests that kNN and SVMs perform at least as well as other algorithms when there is a lot of labeled data for each class of interest [35]. Most studies use the simple bag-of-words document representation.

Automated text summarization dates back to the fifties [21]. The different attempts in this field have shown that human-quality text summarization was very difficult since it encompasses discourse understanding, abstraction, and language generation [30]. Simpler approaches have been explored which consist in extracting representative text-spans, using statistical and/or techniques based on surface domain-independent linguistic analyses. Within this context, summarization can be defined as the selection of a subset of the document sentences which is representative of its content. This is done by ranking document sentences and selecting those with higher score and minimum overlap. This bears a close relationship to text classification.

Text extracting for summarization has been cast in the framework of supervised learning for the first time in the seminal work of [19]. The authors propose a generic summarization model, which is based on a naive Bayes classifier operating on a synthetic representation of sentences. Different authors built on this idea [10, 22].

All these approaches to text classification, ranking and filtering rely on the supervised learning paradigm and require labeling of text spans or documents which is performed manually. Manual tagging is often unrealistic or too costly for building textual resources so that semi-supervised learning is probably well adapted to many information retrieval tasks.

2.2 Semi-supervised learning

The idea of combining labeled and unlabeled data came from the statistician community at the end of the 60's. The seminal paper [12] presents an iterative EM-like approach for learning the parameters of a mixture of two normals with known covariances from unlabeled data. Similar iterative algorithms for building maximum likelihood classifiers from labeled and unlabeled data followed [23, 32]. [13] presented the theory of the EM algorithm, unifying in a common framework many of the previously suggested iterative techniques for likelihood maximization with missing data.

All these approaches are generative, they start from a mixture density model where mixture components are identified to classes and attempt at maximizing the joint likelihood of labeled and unlabeled data. Since direct optimization is usually unfeasible, the EM algorithm is used to perform maximum likelihood estimation. Usually, for continuous variables, density components are assumed to be gaussian especially for performing asymptotic analysis. Practical algorithms may be used for more general settings, as soon as the different statistics needed for EM may be estimated, e.g. for discrete variables, non parametric techniques (e.g. histograms) are often used in practice.

Using likelihood maximization of mixture models for combining labeled and unlabeled data for classification has only been recently rediscovered by the machine learning community and many papers now deal with this subject.

[25] consider a mixture of experts when it is usually assumed that there is a one to one correspondence between classes and components. They propose different models and an EM implementation. [27] propose an algorithm which is a particular case of the general semi-supervised EM described in [24], and present an empirical evaluation for text classification, they also extend their model to multiple components per class. [28] propose a kernel discrimination analysis which can be used for semi-supervised classification. [16] use EM to fill in missing feature values of examples when learning from incomplete data by assuming a mixture model.

There have been considerably fewer works on discriminant semi-supervised approaches. [5] suggests to modify logistic regression, a well known classifier to incorporate unlabeled data. To do so, he maximizes the joint likelihood of labeled and unlabeled data. The co-training paradigm [8] which has been proposed independently is also related to discriminant semi-supervised training. In this approach it is supposed that data x may be described by two modalities which are conditionally independent given the class of x. Two classifiers are used, one for each modality, they operate alternatively as teacher and student.

[11] present an interesting extension of a boosting algorithm which incorporates co-training. The work of [14] also bears similarities with this technique. A transductive support vector machine [33] finds parameters for a linear separator when given labeled data and the data it will be tested on. [18] demonstrates the efficiency of this approach for several text classification tasks. [7] find small improvements on some UCI datasets with simpler variants of transduction. [36] argue both theoretically and experimentally that transductive SVMs are unlikely to be helpful for classification in general.

3 A new discriminant semi-supervised algorithm for classification

In this section, we present a new discriminant algorithm for semi-supervised learning. This algorithm is generic in the sense that it can be used with any classifier which estimates a posteriori class probabilities.

We describe our algorithm in the general framework of the Classification EM (CEM) algorithm. For this we first introduce the general framework of the Classification Maximum Likelihood (CML) approach and the CEM algorithm [9, 24] in section 3.2, we then show in section 3.3 how this framework can lead to a natural discriminant formulation. In the following we introduce briefly the framework of our work.

3.1 Framework

We consider a c-class decision problem and suppose available a set of m unlabeled data $D_u=\{x_i|i=n+1,\ldots,n+m\}$ together with a set of labeled data $D_l=\{(x_i,t_i)|i=1,\ldots,n\}$ where $x_i\in\mathbb{R}^d$ and $t_i=(t_{1i}, \ldots, t_{ci})$ is the class indicator vector for x_i. Data from D_u are

supposed drawn from a mixture of densities with c components $\{C_k\}_{k=1,\ldots,c}$ in some unknown proportions $\{\pi_k\}_{k=1,\ldots,c}$. We suppose that the unlabeled data have an associated missing indicator vector $t_i=(t_{1i}, \ldots, t_{ci})$ for ($i=n+1, \ldots, n+m$) which is a class indicator vector. We further consider that data is partitioned iteratively into c components $\{C_k\}_{k=1,\ldots,c}$. We will denote $\{C_k^{(j)}\}_{k=1,\ldots,c}$ the partition into c clusters computed by the algorithm at iteration j.

3.2 Classification Maximum Likelihood approach

The classification maximum likelihood (CML) approach [31] is a general framework which encompasses many clustering algorithms [9, 24]. It is only concerned with unsupervised learning. In section (3.3) we will extend the CML criteria to semi-supervised learning.

Samples are supposed to be generated by a mixture density:

$$p(x, \Theta)=\sum_{k=1}^{c}\pi_k . f_k(x,\theta_k) \tag{1}$$

where the $\{f_k\}_{k=1,\ldots c}$ are parametric densities with unknown parameters θ_k and π_k is the mixture proportion of k^{th} component. The goal here is to cluster the samples into c components $\{C_k\}_{k=1,\ldots,c}$. Under the mixture sampling scheme, samples x_i are taken from the mixture density p, and the CML criterion is [9, 24]:

$$\log L_{CML}(C,\pi,\theta) = \sum_{k=1}^{c}\sum_{i=n+1}^{n+m} t_{ki}.\log\{\pi_k . f_k(x_i.\theta_k)\} \tag{2}$$

The CEM algorithm [9] is an iterative technique, which has been proposed for maximizing (2), it is similar to the classical EM except for an additional ***C***-step where each x_i is assigned to one and only one component of the mixture. The algorithm is described below.

CEM

Initialization: start from an initial partition $P^{(0)}$

For j^{th} iteration, $j\geq 0$

E-step. Estimate the posterior class probability that x_i belongs to C_k ($i=n+1, \ldots, n+m$, $k=1, \ldots, c$):

$$E[t_{ki}^{(j)} / x_i; C^{(j)},\pi^{(j)},\theta^{(j)}] = \frac{\pi_k^{(j)} . f_k(x_i,\theta_k^{(j)})}{\sum_{k=1}^{c}\pi_k^{(j)} . f_k(x_i,\theta_k^{(j)})} \tag{3}$$

C-step. Assign each x_i to the cluster $C_k^{(j+1)}$ with maximal posterior probability according to $E[t/x]$.

M-step. Estimate the new parameters ($\pi^{(j+1)}$, $\theta^{(j+1)}$) which maximize log $L_{CML}(C^{(j+1)}$, $\pi^{(j)}$, $\theta^{(j)})$

Since the t_{ki} for the labeled data are known, this parameter is either 0 or 1 for examples in D_l, CML can be easily modified to handle both *labeled* and *unlabeled* data [24]. The new criterion - denoted here L_C - becomes:

$$\log L_C(C,\pi,\theta) = \sum_{k=1}^{c}\left\{\sum_{x_i\in C_k}\log\{\pi_k . f_k(x_i,\theta_k)\} + \sum_{i=n+1}^{n+m} t_{ki}.\log\{\pi_k . f_k(x_i.\theta_k)\}\right\} \quad (4)$$

In this expression the first summation is over the labeled samples and the second is over the unlabeled samples.

3.3 Semi-supervised discriminant-CEM

The above generative approach indirectly computes posterior class probabilities $\{p(C_k/x)\}_{k=1,\ldots,c}$ via conditional density estimation. This could lead to poor estimates in high dimensions or when only few data are labeled which is usually the case for semi-supervised learning. On the other hand, in high dimensions, the estimation is carried on a large number of parameters which is time consuming. Since we are dealing with a classification problem, a more natural approach is to directly estimate the posterior class probabilities $p(C/x)$. This is known as the discriminant approach to classification.

In this section, we first rewrite the semi-supervised CML criterion (4) in a suitable form which puts in evidence the role of posterior probabilities.

We then show how it is possible to maximize this likelihood with discriminant classifiers. This leads to a modified CEM algorithm. Using Bayes rule, the CML criterion (4) can be rewritten as:

$$\log L_C(C,\Theta) = \log \widetilde{L}_C(C,\Theta) + \sum_{i=1}^{n+m}\log p(x_i,\Theta) \quad (5)$$

where

$$\log \widetilde{L}_C(C,\Theta) = \sum_{k=1}^{c}\left\{\sum_{x_i\in C_k}\log\{p(C_k / x_i,\pi_k,\theta_k)\} + \sum_{i=n+1}^{n+m} t_{ki}.\log\{p(C_k/ x_i,\pi_k,\theta_k)\}\right\} \quad (6)$$

When using a discriminant classifier with parameters ω to estimate the posterior probabilities of classes, we make no assumption about the marginal distribution $p(x,\Theta)$, therefore the maximum likelihood estimate of ω is the same for L_C than for $\widetilde{L}_C$ [5, 24]. In this case (6) can be written using only the parameters ω of the model:

$$\log \widetilde{L}_C(C,\omega) = \sum_{k=1}^{c}\left\{\sum_{x_i\in C_k}\log\{p_\omega(C_k / x_i)\} + \sum_{i=n+1}^{n+m} t_{ki}.\log\{p_\omega(C_k/ x_i)\}\right\} \quad (7)$$

where $p_\omega(C/x)$ is the estimation of the posterior probability computed by the model.

The maximum likelihood of parameters ω, (7), can be used as a learning criterion for a discriminant classifier. The advantage of this expression is that it is simply expressed upon the output of the classifier which gives suitable properties for the maximization of (7).

In the following we present a new semi-supervised algorithm for discriminant classifiers. With this algorithm, the aim is to maximize (7) with regard to the parameters ω of the classifier.

Because we are interested only in classification, the ***E***-step, which estimates the posterior probabilities using conditional densities in the generative approach is no more necessary. The discriminant-CEM algorithm is summarized below:

Discriminant-CEM

Initialization: Train a discriminant model M estimating the posterior class probabilities with parameters $\omega^{(0)}$ over D_l,
Let $O^{(j)}{}_k$ be the output of the classifier for the k^{th} class at the j^{th} iteration.
For j^{th} iteration, $j \geq 0$

C*-step*. Assign each $x_i \in D_u$ to the cluster $C_k^{(j+1)}$ with maximal posterior probability according to $O^{(j)}{}_k = p_{\omega^{(j)}}(C_k^{(j)} / x_i)$.

M*-step*. Train M over $D_l \cup D_u$ with new parameters $\omega^{(j+1)}$ which maximize $\log \tilde{L}_C(C^{(j+1)}, \omega^{(j)})$.

We have used in our experiments, a stochastic gradient algorithm to maximize (7) in the *M*-step. An advantage of this method is that it requires only the first order derivatives at each iteration. It can easily be proved that this algorithm converges to a local maximum of the likelihood function (7) for semi-supervised training.

The main difference here with the generative method is that instead of estimating class conditional densities, discriminant-CEM algorithm directly attempts to estimate the posterior class probabilities, which is the quantity we are interested in for classification. The above algorithm can be used with any discriminant classifiers which estimate the posterior class probabilities. We have performed experiments using neural networks and support vector machines but for the classification tasks considered here, they did not show any improvement over a simple logistic unit, which in turn performed slightly better than a pure linear classifier.

In the following section we will present results for a series of text classification, filtering and ranking tasks, by using a simple logistic unit with the discriminant-CEM algorithm.

4 Experiments

In the following we will briefly present the data sets we have used (section 4.1) and in section 4.2 we describe our results.

4.1 Data sets

We have used three datasets for text classification tasks: *a*) one of the classification's collection of the NIPS-2001 competition consisting of Web pages. This corpus is composed of 1000 documents where each document is encoded in a fixed vector size. *b*) the e-mail spam classification problem from the UCI repository, this collection is composed of 4601 e-mails, where each e-mail is represented using 57 terms. (features are once again fixed).

For text summarization we have used the Computation and Language (cmp_lg) collection of TIPSTER SMMAC[1]. This corpus is composed of 183 scientific articles. To generate extract-based summaries from the abstract of each article in the collection, we have used the text span alignment method described by [6]. The evaluation is performed by generating a query *q* corresponding to the most frequent words in the training set. To represent each sentence, we considered a continuous version of the features proposed by Kupiec [19]: each sentence *i*, with length *l*(*i*), is represented by a 5 feature vector, $\vec{x}_i$:

$$\vec{x}_i = \{\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5\}$$

where, φ_1 is the normalized sentence length: $\frac{l(i)}{\sum_j l(j)}$, φ_2 is the normalized frequency of cue words in sentence *i*: $\frac{frequency\ of\ cue\ words}{l(i)}$, φ_3 is the normalized number of terms within the query *q* and *i*, φ_4 is the normalized frequency of acronyms in *i*: $\frac{frequency\ of\ acronyms}{l(i)}$ and φ_5 is the same paragraph feature as in [19].

In all cases, the data was randomly split into a training and a test set whose size was respectively 1/3 and 2/3 of the available data.

4.2 Results

For text summarization a compression ratio must be defined for extractive summaries. For the cmp_lg collection we followed the SUMMAC evaluation by using 10% compression ratio. For evaluation we compared the extract of a system with the desired summary and used the average precision measure to evaluate our system. Where the precision is defined as:

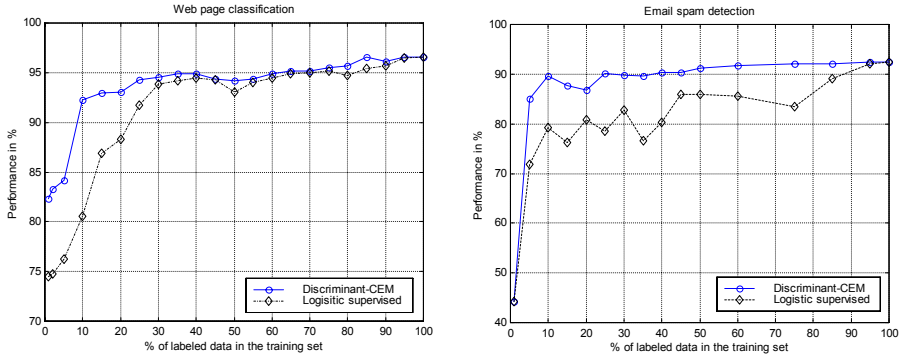
$$\text{Precision} = \frac{\#\text{ of sentences extracted by the system which are in the target summaries}}{\text{total }\#\text{ of sentences extracted by the system}}$$

[1] http://www.itl.nist.gov/iaui/894.02/related_projects/tipster_summac/cmp_lg.html

For Text classification, we followed the NIPS's workshop evaluation which considered the Percentage of Good Classification (PGC) defined as:

$$\text{PGC} = \frac{\#\text{ of examples of the test set well classified by the system}}{\#\text{ of examples in the test set}}$$

For our experiments we used a logistic unit as the baseline classifier. Figure 1 and 2 show performance on the test sets respectively for text classification and text summarization tasks. These figures plot a score for different ratio of labeled-unlabeled data in the training set. On the *x*-axis, 5% means that 5% of data in the training set were labeled for training, the 95% remaining being used as unlabeled training data. For comparison, we have also performed tests with the logistic classifier trained in a supervised way using the same *x*% labeled data in the training set.



(*a*) (*b*)

Figure 1: Web Pages classification (*a*) and e-mail Spam detection (*b*) - Performance of two classifiers with respect to the ratio *x* of labeled data in the training set. The classifier is a logistic unit trained with labeled data in a supervised scheme (dashed bottom curves) and using the semi-supervised discriminant-CEM algorithm (solid top curves).

For both classification tasks, the logistic classifier trained only on *x*% labeled data performs well but is clearly below the discriminant-CEM algorithm particularly in the regions of interest where the ratio of labeled data is small. For example, for web pages (figure 1-*a*) at 10% labeled data, semi supervised training reduces the classification error by more than 12% compared to the same classifier trained without unlabeled data. This shows empirically that unlabeled data do indeed contain relevant information and that the semi-supervised learning algorithm proposed here allows extracting part of this information.

For text summarization, we have also compared in figure 2, discriminant-CEM to the generative-CEM algorithm presented in section 3.2. For the latter, we assume that the conditional density functions $\{f_k\}_{k=1,\ldots,c}$ are normal distributions.

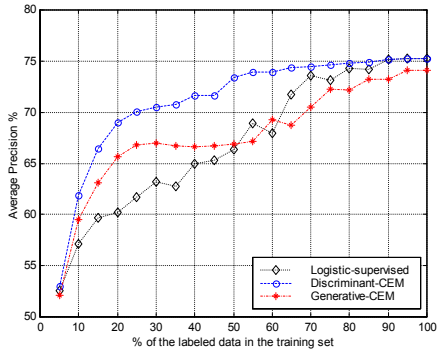


Figure 2: Average precision of 3 trainable summarizers with respect to the ratio of labeled sentences in the training set for the cmp_lg collection.

This comparison was not possible for text classification, due to the numerical inversion problems of covariance matrices.

This problem is frequently seen, with sparse matrices of document representations in high dimensions. Discriminant-CEM uniformly outperforms generative-CEM in all regions for text summarization. This is particularly clear for SUMMAC cmp_lg, which is a small document set. In this case, the discriminant approach is clearly superior to the generative approach which suffers from estimation problems.

Table 1 compares the Kupiec et al. summarizer system with the generative and discriminant CEM algorithms, all trained in a fully supervised way on the whole training set.

Table 1. Comparison between kupiec et al.'s summarizer system and discriminant and generative CEM algorithms for the cmp_lg collection. All classifiers are trained in a fully supervised way.

System	Average Precision (%)	PGC(%)
Kupiec's system	61,83	63,48
Generative-CEM	74,12	74,79
Discriminant-CEM	75,26	76,92

The two CEM classifiers allow approximately 10% increase both in average precision and in accuracy over Kupiec et al's system. Another interesting result is that both discriminant and generative CEM trained on semi-supervised learning scheme (using 10% of labeled sentences together with 90% of unlabeled sentences in the training set) gave similar performances to the Kupiec et al.'s summarizer system fully supervised.

5 Conclusion

We have introduced a new discriminant algorithm for training classifiers in presence of labeled and unlabeled data. This algorithm has been derived in the framework of CEM algorithms and is pretty general in the sense that it can be used with any discriminant classifier. We have provided experimental analysis of the proposed method for text classification and text summarization with regard to ratio of labeled data in the training set, and we have shown that the use of the unlabeled data for supervised learning can indeed increase the classifier accuracy. We have also compared discriminant and generative approaches to semi-supervised learning and the former has been found clearly superior to the latter especially for small collections.

References

1. Amini, M.-R., Gallinari P.: Learning for Text Summarization using labeled and unlabeled sentences. Proceedings of the 11th International Conference of Artificial Neural Networks, (2001), 1177-1184.
2. Amini, M.-R., Gallinari P.: Automatic Text Summarization using Unsupervised and Semi-supervised Learning. Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery, (2001) 16-28.
3. Amini, M.-R., Gallinari P.: Semi-supervised Logistic Regression. Proceedings of the 15th European Conference on Artificial Intelligence, (2002), to appear.
4. Amini, M.-R., Gallinari P.: The Use of the labeled data to Improve Supervised Learning for Text Summarization. Proceedings of the 25th International ACM SIGIR, (2002), to appear.
5. Anderson, J. A., Richardson, S.C.: Logistic Discrimination and Bias correction in maximum likelihood estimation. Technometrics, Vol. 21. (1979) 71-78.
6. Banko, M. Mittal V., Kantrowitz, M., Goldstein, J.: Generating Extraction-Based Summaries from Hand-written done by text alignment. Pac. Rim Conf. On Comp. (1999).
7. Bennet, K., Demirez, A.: Semi-supervised Support Vector machines. In Kearns, Solla, and Cohn, editors. Advances in Neural Information Processing Systems 11. MIT Press (1998) 368-374
8. Blum, A., Mitchell, T.: Combining Labeled and unlabeled Data with Co-Training. Proceedings of the Conference on Computational Learning Theory (1998) 92-100
9. Celeux, G., Govaert, G.: A Classification EM algorithm for clustering and two stochastic versions. Computational Statistic and Data Analysis Vol. 14 (1992) 351-332.
10. Chuang, W.T., Yang, J.: Extracting sentence segments for text summarization: a machine learning approach. Proceedings of the 23rd ACM SIGIR. (2000) 152-159.
11. Collins, M., Singer, Y.: Unsupervised models for named entity classification. In Proceedings of EMNLP (1999)
12. Day N. E., Estimating the components of a mixture of normal distributions. Biometrika, Vol. 56, N° 3. (1969) 463-474.
13. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via EM algorithm. Journal of the Royal Statistical Society, Vol. B, n°39 (1977) 1-38
14. De Sa, V.R.: Learning Classification with Unlabeled Data. Neural Information Processing Systems, Vol. 6 (1993) 112-119.

15. Dumais, S. T., Platt J., Heckerman, D., Sahami M.: Inductive learning algorithms and representations for text categorization. CIKM. (1998) 148-155.
16. Ghahramani, Z., Jordan M.I.: Supervised learning from incomplete data via EM approach. Advances in Neural Information Processing Systems, Vol. 6, (1994) 120-127.
17. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. Tenth European Conference in Machine Learning (1998) 137-142.
18. Joachims, T.: Transductive inference for text classification using support vector machines. Proceedings of sixteenth International Conference on Machine Learning (1999) 200-209.
19. Kupiec J., Pderson J., Chen F.A.: Trainable Document Summarizer. Proceedings of the 18th ACM SIGIR (1995) 68-73.
20. Lewis, D. D.: Naive (Bayes) at forty: The independence assumption in information retrieval. Tenth European Conference in Machine Learning (1998) 4-15.
21. Luhn, P.H.: Automatic creation of literature abstracts. IBM Journal (1958) 159-165.
22. Mani, I., Bloedorn, E.: Machine Learning of Generic and User-Focused Summarization. Proceedings of the Fifteenth National Conference on AI. (1998) 821-826.
23. McLachlan, G.J.: Iterative reclassification procedure for constructing asymptotically optimal rule of allocation in discriminant analysis. Journal of the American Statistical Association. Vol. 70, N° 350, (1975) 365-369.
24. McLachlan, G.J.: Discriminant Analysis and Statistical Pattern Recognition. John Willey and Sons, New York (1992)
25. Miller, D., Uyar, H.: A Mixture of Experts classifier with learning based on both labeled and unlabeled data. Advances in Neural Information Processing Systems 9 (1996) 571-577
26. Mosteller, F., Wallace, D. L.: Inference and disputed authorship: The Federalist. Massachusetts: Addison-Wesley, (1964).
27. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text Classification from Labeled and Unlabeled Documents using EM. Machine Learning, Vol 39, N° 2/3, 103-104 (2000).
28. Roth V., Steinhage, V.: Nonlinear Discriminant Analysis using Kernel Functions. Advances in Neural Information Processing Systems, Vol. 12, (1999).
29. Schapire, R. E., Singer, Y.: BoosTexter: A Boosting-based system for text categorization. Machine Learning, Vol. 39, N° 2/3. (2000) 135-168.
30. Sparck Jones, K.: Discourse modeling for automatic summarizing. Technical report 29D, Computer laboratory, university of Cambridge. (1993).
31. Symons, M.J.: Clustering criteria and Multivariate Normal Mixture. Biometrics. Vol. 37 (1981) 35-43.
32. Titterington, D.M.: Updating a diagnostic system using unconfirmed cases. Applied Statistics, Vol. 25, N° 3, (1976) 238-247.
33. Vapnik, V.: Statistical learning theory. John Wiley, New York.
34. Wiener, E., Pederson, J. O., Weigend, A. S.: A neural network approach to topic spotting. Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval. (1995) 317-332.
35. Yang Y: An evaluation of statistical approaches to text categorization. Information Retrieval, Vol. 1, N° 2/3. (1999) 67-88.
36. Zhang, T., Oles, F.J.: A probability analysis on the value of unlabeled data for classification problems. Proceedings of the Seventeenth International Conference on Machine Learning (2000) 1191-1198.