Pattern Recognition Letters 32 (2011) 297–304

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Learning aspect models with partially labeled data

Anastasia Krithara [a,b,*], Massih R. Amini [b,c], Cyril Goutte [c], Jean-Michel Renders [a]

[a] Xerox Research Centre Europe, 6, Chemin de Maupertuis, 38240 Meylan, France
[b] University Pierre et Marie Curie, 4, Place Jussieu, 75252 Paris cedex 05, France
[c] National Research Council Canada, 283, Bd. Alexandre-Taché, QC, Canada J8X 3X7

ARTICLE INFO

Article history:
Received 29 October 2009
Available online 19 September 2010
Communicated by T.K. Ho

Keywords:
Semi-supervised learning
Aspect models
Document categorization

ABSTRACT

In this paper, we address the problem of learning aspect models with partially labeled data for the task of document categorization. The motivation of this work is to take advantage of the amount of available unlabeled data together with the set of labeled examples to learn latent models whose structure and underlying hypotheses take more accurately into account the document generation process, compared to other mixture-based generative models. We present one semi-supervised variant of the Probabilistic Latent Semantic Analysis (PLSA) model (Hofmann, 2001). In our approach, we try to capture the possible data mislabeling errors which occur during the training of our model. This is done by iteratively assigning class labels to unlabeled examples using the current aspect model and re-estimating the probabilities of the mislabeling errors. We perform experiments over the `20Newsgroups`, `WebKB` and `Reuters` document collections, as well as over a real world dataset coming from a Business Group of Xerox and show the effectiveness of our approach compared to a semi-supervised version of Naive Bayes, another semi-supervised version of PLSA and to transductive Support Vector Machines.

1. Introduction

One of the major challenges in many supervised Machine Learning (ML) tasks is to build suitable annotated databases.

In many textual information access applications, skilled humans are needed in order to annotate the data, especially in technical domains. With the explosion of information that occurred during the last years, the annotation of documents has become a bottleneck for supervised learning. The labeling process is so time-consuming that just a part of the available data can be labeled. On the other hand, huge amounts of unlabeled data are available and easy to obtain. This has sparked the interest of the ML community to design new algorithms able to learn from partially labeled training sets. These algorithms, referred to as semi-supervised learning (SSL) algorithms in the literature, rely on the assumption that unlabeled examples carry some useful information about the problem we try to solve.

At this point, the question that arises is *if*, and *under which circumstances*, unlabeled data can be helpful. The research already conducted to answer this question has demonstrated, with theoretical and experimental results, that unlabeled examples can, under some assumptions, help improve classification performance. The most common assumptions are:

- the *smoothness* assumption: nearby examples are likely to be in the same class;
- the *cluster* assumption: the decision boundary should be found in low-density regions (i.e., between high density clusters); and
- the *manifold* assumption: examples which belong to the same manifold have the same class.

Following these studies, several practical algorithms have been proposed, many of them motivated by textual information access tasks such as document classification or summarization (Nigam et al., 2000; Amini and Gallinari, 2002; Joachims, 1999).

Some SSL methods rely on a generative model (Nigam et al., 2000; Amini and Gallinari, 2003). They usually implement a local independence assumption (similar to the Naive Bayes *assumption*), which is unlikely to be met in practice. In addition, some simpler models (such as the Naive Bayes *model*) assume that an observation is generated in its entirety from the class it belongs to. This makes it inconvenient to model data that may comprise several aspects, such as textual documents which potentially cover different topics. This has led to the development of aspect models (Hofmann, 2001), which can take into account data with multiple facets. In this class of models, observations are generated by a mixture of aspects, or topics, each of which being a distribution over the basic features of the observations (such as words in a document). Interestingly, these models also account for non-trivial application-dependent phenomena. When modeling textual content, for example, they take into account linguistic properties

* Corresponding author. Address: National Centre for Scientific Research, "Demokritos", Athens, Greece. Tel.: +302106503204; fax: +30 2106532175.
E-mail address: akrithara@iit.demokritos.gr (A. Krithara).

0167-8655/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2010.09.004

such as synonymy (different terms with the same meaning) and polysemy (different meanings of the same term). Both may have a crucial influence on the modeling of the relationship between documents.

To the best of our knowledge, there has been little effort so far to extend these aspect models to the semi-supervised framework. In this paper, we review two semi-supervised variants of the *probabilistic latent semantic analysis* (PLSA) model originally proposed by Hofmann (2001) in the unsupervised learning framework. In our first model, additional *fake* labels are introduced to account for unlabeled data (Gaussier and Goutte, 2005). The motivation is to avoid predicting arbitrary labels for mainly unlabeled components, i.e., the components with few or no labeled examples and many unlabeled observations. The second algorithm iteratively computes class labels for unlabeled data and estimates the class labeling errors using a *mislabeling error model* (Amini and Gallinari, 2003). We assume that the labeling errors made by the generative model for unlabeled data come from a stochastic process and that these errors are inherent to semi-supervised learning. The idea is then to characterize this stochastic process in order to reduce the labeling errors computed by the classifier for unlabeled data in the training set.

This article is organized as follows. A brief review of the literature on semi-supervised learning and mislabeling error models is given in Section 2. Section 3 discusses semi-supervised generative models for classification. First, the supervised version of PLSA (Hofmann, 2001; Gaussier et al., 2002) is presented (Section 3.1), then a semi-supervised variant is proposed in Section 3.2. Section 4 describes experiments illustrating the performance of the above model on several benchmarks and comparing them to some existing semi-supervised algorithms, as well as the well-known transductive version of Support Vector Machines (Joachims, 1999). We conclude and discuss future work in Section 5.

2. Related work

As producing labeled training sets for supervised learning is often expensive and difficult, important issues are that (*a*) the labeling process is so time consuming that just a part of the available data can be labeled, or (*b*) the actual class labels are usually subject to error. Sections 2.1 and 2.2 address these two points and Section 2.3 gives some context on aspect models.

2.1. Learning with partially labeled training data

Semi-supervised learning was studied in statistics in the late 60's (Fralick, 1967). Since the mid-90s, the Machine Learning (ML) community has shown real interest in developing efficient SSL algorithms, mostly due to applications in text classification and natural language processing problems (Blum and Mitchell, 1998; Nigam et al., 2000).

SSL algorithms are typically *inductive*: they learn a decision rule based on both labeled and unlabeled training examples, which can later be applied to new unseen observations. Early SSL ideas were based on the principle of *self-training*: a classifier is trained on the labeled examples, and the resulting decision function is used in order to annotate some unlabeled examples. The additional annotation is used to re-train a classifier and the procedure iterates until the predicted labels do not vary anymore. In the case where the cluster assumption is satisfied, the self-training principle tries to find a decision boundary which lies in low density regions (Amini et al., 2009).

Note that SSL differs from *transductive* learning as introduced by Vapnik (1998) in that the transductive learning attempts to annotate unlabeled test examples without inferring a general decision rule.

Our work deals with a self-training SSL algorithm designed under the generative framework. Many generative SSL methods rely on mixture models (McLachlan, 1992), following the *cluster assumption*. The mixture model characterizes both the input distribution and the labeling process. Labeled and unlabeled examples are used jointly to estimate the mixture components, typically using the Expectation-Maximization (EM) algorithm (Dempster et al., 1977), and the labeled examples are used to estimate component labeling probabilities. As a consequence, the decision boundary falls in between clusters of data, in low density regions. This approach was introduced by Miller and Uyar (1997), and applied to document classification by Nigam et al. (2000). Note that most theoretical studies of SSL were developed under the generative framework, focusing on the role of unlabeled examples in the learning process (O'Neill, 1978; Castelli and Cover, 1995, 1996). On the other hand, discriminative approaches to SSL focus on directly estimating the decision boundary between classes, without specifically implementing the cluster assumption (Grandvalet and Bengio, 2005; Vittaut et al., 2002). Although discriminative approaches are known to be asymptotically better than generative algorithms, the latter may be preferable when the training data is limited (Ng and Jordan, 2002).

2.2. Learning with misclassified training data

The difficulty of the annotation process may also lead to mislabeling of examples. This has motivated work in the pattern recognition community since the 70s, on the topic of learning in the presence of labeling noise. These studies distinguish between *random* and *non-random* imperfect supervisions. The latter corresponds to the situation in which the probability of mislabeling an example depends on its feature vector (Lachenbrunch, 1974; Titterington, 1989; Ambroise and Govaert, 2000), and it has been much less studied than the random case.

In *random imperfect supervision*, the probability of misclassification of an observation is independent of its feature vector, e.g. when training data is labeled on the basis of machine output. This setting has been studied extensively in the framework of generative models (McLachlan, 1972; Chittineni, 1980; Krishnan and Nandy, 1987, 1988; Lawrence and Schlkopf, 2001). For each example in the training set, these approaches estimate the probability of the incorrect label given the correct one, and use these probabilities while updating the model parameters. Note that a theoretical analysis of the importance of mislabeled training data (Chhikara and McKeon, 1984) showed that ignoring mislabeling while training a classifier can degrade classification performance.

For binary classification, Amini and Gallinari (2003) introduced a SSL method which takes mislabeling into account. Although it relies on *random imperfect supervision*, it does not assume that labeling errors come from *manual* labeling of the data, contrary to previous techniques for learning from imperfect labels. Instead, it assumes that the mislabeling errors arise from the classification itself and it uses a *mislabeling error model* to correct them, which is incorporated into a logistic regression classifier. The classifier is first trained on the labeled part of the training set and it iteratively assigns class labels to unlabeled training examples. These newly labeled examples, together with the labeled part of the training set, are then used to train a new logistic classifier. At each labeling step, the semi-supervised learning system acts as an *imperfect supervisor* on the unlabeled training examples.

In our work, we extend that approach to the more general multiclass case by incorporating a mislabeling error model into a semi-supervised PLSA algorithm for text categorization. In the next section, we give some context on aspect models related to PLSA.

2.3. Aspect models

Aspect models are based on the idea that documents can be modeled as a mixture of topics. They differ in the statistical assumptions they impose on the model. The first widely used aspect model is the probabilistic latent semantic analysis (PLSA) introduced by Hofmann (2001). In PLSA, documents are represented as random mixtures over latent topics, and each topic is characterized by a multinomial distribution over words. PLSA is a generative model of the training collection, but not of arbitrary previously unseen documents. To fix that, Blei et al. (2003) introduced the latent Dirichlet allocation (LDA), which introduces a Dirichlet prior on the mixture weights. The generative aspect model (GAM) (Minka and Lafferty, 2002) extends LDA by varying word probabilities stochastically across documents. These models are all *generative* for some documents[1]: they specify a simple probabilistic procedure by which theses documents can be generated. They aim at providing a better representation of text documents, by implementing some features of natural language such as the fact that one textual unit may be generated by several topics (polysemy).

This paper is the continuation of, and builds on earlier work on SSL for PLSA (Krithara et al., 2006, 2008). The focus of Krithara et al. (2006), however, was on reducing the annotation burden by relying on SSL using *fake labels* on the unlabeled documents, and combining that with active learning. The model described in (Krithara et al., 2008) also combines SSL with a mislabeling error model. The current paper, however, extends and improves that model by decoupling labeling probabilities for labeled and unlabeled examples. In addition, it provides a more extensive experimental evaluation, investigating additional issues such as the behavior for very low annotation ratios.

3. Semi-supervised generative models for document categorization

This section presents probabilistic frameworks for document categorization. We describe PLSA and its use for supervised classification (Section 3.1), and introduce an extension to partially labeled data, relying on a mislabeling error model (Section 3.2).

We assume that we have a collection of documents $\mathcal{X} = \{x_1, \ldots, x_N\}$, which consists of two subsets, one of labeled data, $\mathcal{X}_l$, and one of unlabeled data, $\mathcal{X}_u$, such that $\mathcal{X} = \mathcal{X}_l \cup \mathcal{X}_u$. Each document $x \in \mathcal{X}$ is represented by a vector of feature frequencies $\boldsymbol{x} = [n(w,x)]_{w \in \mathcal{W}}$, where w is an observable feature, typically a word, from set $\mathcal{W} = \{w_1, \ldots, w_{|\mathcal{W}|}\}$, and $n(w,x)$ is the number of times w was observed in x. All documents x from $\mathcal{X}_l$ have a class label $y \in \mathcal{C} = \{y_1, \ldots, y_K\}$, while the class label is unknown for documents from $\mathcal{X}_u$.

3.1. Probabilistic latent semantic analysis

PLSA (Hofmann, 2001) characterizes the generation of co-occurrences of words and documents as a probabilistic mixture model. This model, also known as the aspect model (Saul and Pereira, 1997), associates an unobserved latent variable (*aspect* or *topic*) $\alpha \in \mathcal{A} = \{\alpha_1, \ldots, \alpha_L\}$ to each observation corresponding to the occurrence of a feature $w \in \mathcal{W}$ within a document $x \in \mathcal{X}$. Topics can coincide with classes or, in another setting, a class can contain several *aspects*. In addition, mixture components are conditionally-independent multinomial distributions, $P(w,x|\alpha) = P(w|\alpha)P(x|\alpha)$. Let us now define the training sets in terms of pairs of co-occurrences $q = (w,x)$ rather than examples x. Using the examples from set $\mathcal{X}_l$ (resp. $\mathcal{X}_u$), we form a labeled (resp. unlabeled) training set of co-occurrences $\mathcal{Q}_l$ (resp. $\mathcal{Q}_u$). For the labeled set $\mathcal{Q}_l$, the label of each $q \in \mathcal{Q}_l$ is the label of the document x contained in q.

Although originally proposed in an unsupervised setting, PLSA is easily extended to partially labeled data with the following generative process:

- Pick an example x with probability $P(x)$,
- Choose a latent variable α according to conditional probability $P(\alpha|x)$
- Sample a feature w with probability $P(w|\alpha)$
- For labeled data: Pick the class y according to probability $P(y|\alpha)$.

This generation scheme results in either unlabeled pairs (x,w) or labeled triples (x,w,y), while the latent variables α are discarded. Fig. 1 depicts the generative processes for Naive Bayes (Nigam et al., 2000) and PLSA. Both models generate words independently from each other (the bag-of-words assumption), but whereas Naïve Bayes generates all words from the same topic associated to class y, PLSA resamples the topic each time it generates a word.

The generation of an unlabeled pair (w,x) has the following probability:

$$P(w,x) = P(x)\sum_{\alpha\in\mathcal{A}} P(w|\alpha)P(\alpha|x) = \sum_{\alpha\in\mathcal{A}} P(w|\alpha)P(x|\alpha)P(\alpha), \quad (1)$$

while labeled triples have probability:

$$P(w,x,y) = P(x)\sum_{\alpha\in\mathcal{A}} P(w|\alpha)P(\alpha|x)P(y|\alpha). \quad (2)$$

This model overcomes some simplifying assumptions of Naive Bayes in two important ways. First, it relaxes the assumption that a class y is associated to a single topic. In PLSA, the number of topics $|\mathcal{A}|$ may be larger than the number of classes K. The second and crucial difference is that in Naive Bayes, all words must be generated from the same topic. This requires the use of clever smoothing strategies to counter the fact that words unrelated to a topic may appear by coincidence in a document from that topic. On the other hand, in PLSA, a topic is drawn independently from $P(\alpha|x)$ each time a new feature is generated in an example. This provides a much more natural way to handle unusual words or multitopicality.

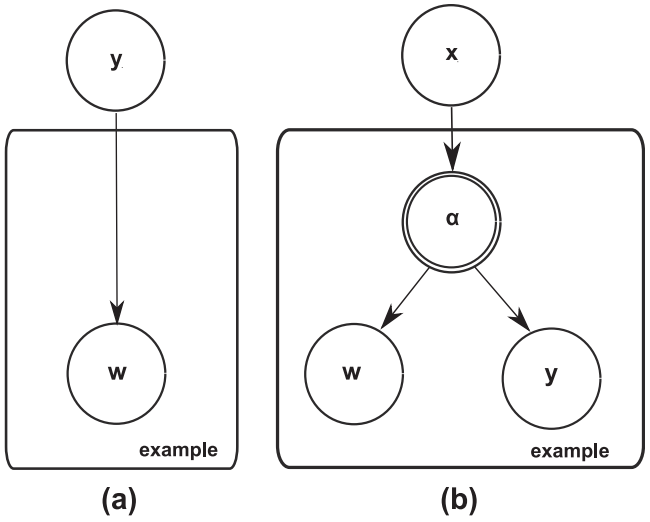

Fig. 1. Graphical model representations of (a) Naive Bayes and (b) PLSA. The topic latent variable, α, is double circled.

[1] Only documents from the training collection for PLSA, potentially all documents for LDA and GAM.

3.2. Semi-Supervised PLSA with a mislabeling error model

In this section we introduce a semi-supervised variant of PLSA incorporating a misclassification error model. We assume that the errors made when labeling the unlabeled data according to the generative model come from a stochastic process, and that these errors are inherent to semi-supervised learning algorithms. By characterizing and learning this stochastic process, we will reduce the labeling errors made by the classifier over unlabeled data.

Assume that for each unlabeled document $x \in \mathcal{X}_u$, there exists a perfect, true label y, and an imperfect label $\tilde{y}$, estimated by the classifier. Assuming that $\tilde{y}$ depends on y, we model these mislabeling errors by:

$$\forall (k,h) \in \mathcal{C} \times \mathcal{C}, \beta_{kh} = P(\tilde{y} = k | y = h), \tag{3}$$

subject to the constraint that $\forall h, \sum_k \beta_{kh} = 1$, i.e. β is column-stochastic.

Fig. 2 gives a graphical representation of this model for both labeled and unlabeled data. The underlying generative process for this model is:

- ▹ Pick a document x with probability $P(x)$,
- ▹ Choose a latent variable α according to conditional probability $P(\alpha|x)$
- ▹ Sample a feature w with probability $P(w|\alpha)$
- ▹ Generate the latent *true* class y with probability $P(y|\alpha)$
- ▹ Pick the imperfect class label $\tilde{y}$ with probability $\beta_{\tilde{y}|y} = P(\tilde{y}|y)$

The values of $P(y|\alpha)$ depend on the value of latent topic variable α. For labeled training data, the repartition of these variables α in different classes is also known. So, at the initialization step the probabilities $P(y|\alpha)$ are forced to zero for latent topic variables α which do not belong to the class y. On the contrary, for unlabeled training data, the probabilities $P(y|\alpha)$ are left free and we designate these degrees of freedom by $\tilde{p}(y|\alpha)$; they are associated only with unlabeled data. The values $\tilde{p}(y|\alpha)$ are first initialized at random and are then estimated during the learning phase.

With this new graphical model, the joint probability for an occurrence from an unlabeled document, $q = (w,x) \in \mathcal{Q}_u$, and its imperfect class label estimated by the classifier is:

$$\forall q \in \mathcal{Q}_u, P(w,x,\tilde{y}) = P(x) \sum_{\alpha \in A} P(w|\alpha) P(\alpha|x) \sum_{y \in \mathcal{C}} \beta_{\tilde{y}|y} \widetilde{P}(y|\alpha).$$

The corresponding model parameters[2] are

$$\Phi = \{P(\alpha|x), P(w|\alpha), \beta_{\tilde{y}|y}, \widetilde{P}(y|\alpha) : x \in \mathcal{X}, w \in \mathcal{W}, \alpha \in \mathcal{A}, y \in \mathcal{C}, \tilde{y} \in \mathcal{C}\}$$

and are estimated by maximizing the complete data log-likelihood

$$\mathcal{L}_2 = \sum_{x \in \mathcal{X}_l} \sum_w n(w,x) \log \sum_\alpha P(x) P(w|\alpha) P(\alpha|x) P(y|\alpha) + \sum_{x \in \mathcal{X}_u} \sum_w n(w,x) \log \sum_\alpha P(x) P(w|\alpha) P(\alpha|x) \sum_y \beta_{\tilde{y}|y} \widetilde{P}(y|\alpha), \tag{4}$$

using an iterative EM algorithm, as sketched out in Algorithm 1. Parameters are initialized at random. Then, at each iteration t, the `E-step` estimates the latent topic posteriors for labeled and unlabeled data, given parameters $\Phi^{(t)}$[3]:

$$\pi_\alpha(w,x,y) = \frac{P(\alpha|x)P(w|\alpha)P(y|\alpha)}{\sum_\alpha P(\alpha|x)P(w|\alpha)P(y|\alpha)}, \tag{5}$$

$$\tilde{\pi}_\alpha(w,x,\tilde{y}) = \frac{P(\alpha|x)P(w|\alpha)\sum_y \widetilde{P}(y|\alpha)\beta_{\tilde{y}|y}}{\sum_\alpha P(\alpha|x)P(w|\alpha)\sum_y \widetilde{P}(y|\alpha)\beta_{\tilde{y}|y}}. \tag{6}$$

[2] As we use a hard assignment of components α to classes y, this is reflected in set binary values for $P(y|\alpha)$, which are therefore not part of the learnable parameters.

[3] For brevity, we omit the superscript $^{(t)}$ on the right hand side of the EM equations.

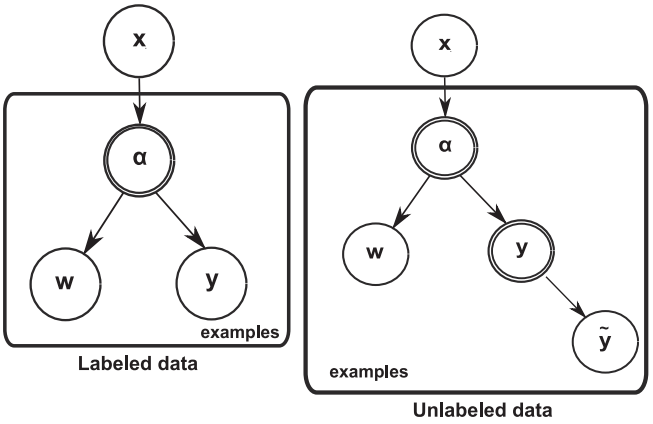

Fig. 2. Graphical model representation of the semi-supervised PLSA with a mislabeling error model, for labeled (left) and unlabeled (right) documents.

Algorithm 1: ssPLSA with mislabeling error model

Input
- ◇ A set of partially labeled data $\mathcal{X} = \mathcal{X}_l \cup \mathcal{X}_u$,
- ◇ Training sets $\mathcal{Q}_l$ and $\mathcal{Q}_u$ formed from $\mathcal{X}_l$ and $\mathcal{X}_u$.

Initialization
- ◦ Random initial model parameters $\Phi^{(0)}$,
- ◦ Run a standard PLSA algorithm for estimating the initial $\tilde{y}$,
- ◦ $t \leftarrow 0$.

repeat
- • `E-step`: Compute latent class posteriors (Eqs. (5) and (6)) given $\Phi^{(t)}$
- • `M-step`: Estimate the new model parameters $\Phi^{(t+1)}$ maximizing the complete-data log-likelihood (Eqs. (7)–(10))
- • $t \leftarrow t + 1$

until *convergence of the complete-data log-likelihood (4)*;

Output: A generative classifier with parameters $\Phi^{(t)}$.

The `M-step` updates parameters by maximizing the likelihood (4) given the above posteriors:

$$P^{(t+1)}(\alpha|x) \propto \sum_w n(w,x) \times \begin{cases} \pi_\alpha(w,x,y), \text{for} x \in X_l, \\ \tilde{\pi}_\alpha(w,x,\tilde{y}), \text{for} x \in X_u, \end{cases} \tag{7}$$

$$P^{(t+1)}(w|\alpha) \propto \sum_{x \in X_l} n(w,x)\pi_\alpha(w,x,y) + \sum_{x \in X_u} n(w,x)\tilde{\pi}_\alpha(w,x,\tilde{y}), \tag{8}$$

$$\beta^{(t+1)}_{\tilde{y}|y} \propto \sum_w \sum_{x \in X_u} n(w,x) \sum_\alpha \tilde{\pi}_\alpha(w,x,\tilde{y}), \tag{9}$$

$$\widetilde{P}^{(t+1)}(y|\alpha) \propto \sum_w \sum_{x \in X_u} n(w,x)\tilde{\pi}_\alpha(w,x,\tilde{y})\beta^{(t)}_{\tilde{y}|y}. \tag{10}$$

These updated parameters form $\Phi^{(t+1)}$ and the `E-step` and `M-step` are iterated until convergence of the likelihood (4). The complexity of this algorithm is $O(L \times K \times M)$, with M the number of co-occurrences with non-zero frequency.

At this point, the distinction between the above algorithm (referred to as PLSA with a *m*islabeling *e*rror *m*odel and *s*oft clustering, or *ssPLSA-mems*, in the following) and the one presented in (Krithara et al., 2008) becomes clearer. The latter (PLSA with a *m*islabeling *e*rror *m*odel and *h*ard clustering, or *ssPLSA-memh*) also uses a mislabeling error. However, the main difference lies in the fact that *it does not perform soft clustering for the unlabeled data as described above*. Instead, the $P(y|\alpha)$ values are fixed and kept constant for

both labeled and unlabeled data. Experiments presented in the next section will show that the freedom we give to the unlabeled data by doing soft clustering is really beneficial, especially when the ratio of labeled-unlabeled data is very low.

4. Experiments

In order to evaluate the algorithm proposed in the previous sections, we performed experiments on four different datasets. We will first describe these datasets (Section 4.1) and the evaluation measures (Section 4.2), before presenting the results we obtained in Section 4.3.

4.1. Data sets

In our experiments, we used two collections from the CMU World Wide Knowledge Base (WebKB) project - `WebKB`, `20Newsgroups`,[4] the widely used text collection of `Reuters` 21578 and an additional dataset coming from a Xerox Business Group (the `XLS` collection). Note that although we focus on document classification, the algorithms described in the previous sections could also be used for different applications, such as image classification.

For all four datasets, we ran four semi-supervised algorithms:

- semi-supervised Naive Bayes (ssNB, Nigam et al., 2000),
- semi-supervised PLSA with fake labels (ssPLSA-fake, Gaussier and Goutte, 2005; Krithara et al., 2006),
- semi-supervised PLSA with mislabeling error model and hard clustering (ssPLSA-memh, Krithara et al., 2008),
- semi-supervised PLSA with mislabeling error model and soft clustering (ssPLSA-mems, described in Section 3.2)

In addition, we ran the Transductive Support Vector Machine (TSVM) algorithm (Joachims, 1999) on all but the Xerox dataset.

The `20Newsgroups` dataset is a standard document collection for text categorization. It contains 20 000 messages collected from 20 Usenet newsgroups.

The `WebKB` dataset contains web pages gathered from 4 different university computer science departments, and divided into seven categories. In this paper, we use all pages from the four most commonly used categories (Nigam et al., 2000): student, faculty, course and project, all together containing 4196 documents.

The `Reuters` dataset consists of 21578 articles and 90 topic categories from the Reuters newswire. We selected the documents which belong to only one category. In addition we only kept the seven categories which contain at least 100 documents. This yields a collection of 4381 documents in 7 different categories.

Finally, the `XLS` dataset contains 54,770 documents, 20,000 in the training set and 34,770 in the test set. In the domain of litigation services, this is a typical corpus that is processed in an e-discovery phase and consists of corporate documents more or less related to a particular case. The documents consist of approximately 35% emails, 20% Microsoft Word documents; 15% Microsoft Excel documents, 10% Microsoft Power point documents and 10% PDF. There are also a significant part of scanned OCR-ized documents, whose textual content is very noisy. The task is a binary categorization problem: we want to classify the documents as *Responsive* or *Non-Responsive* to a particular given case. This is a rather subtle and complex task, as the boundary of what is responsive is fuzzy and subject to multiple interpretations (there are typically a dozen of annotators, with different understandings of the requirements). The two categories are balanced (50%/50%).

Table 1 summarizes the characteristics of these datasets.

Table 1
Characteristics of the datasets

Dataset	`20Newsgroups`	`WebKB`	`Reuters`	`XLS`
Collection size	20 000	4196	4381	54 770
# of classes, K	20	4	7	2
Vocabulary size, $\lvert\mathcal{W}\rvert$	38 300	9400	4749	10 000
Training set size, $\lvert X_l \cup X_u\rvert$	16 000	3257	3504	20 000
Test set size	4000	839	876	34 770

All four datasets were preprocessed as follows:

- Email tags as well as other non-alphanumeric terms were removed,
- All documents were tokenized on white space and punctuation,
- Tokens with a document frequency less than 5 were discarded,
- A total of *608* stopwords from the `CACM` stoplist[5] were filtered out.

No other form of processing (stemming, multiword recognition, etc.) was used on the documents.

4.2. Evaluation

In order to evaluate the performance of the various algorithms used in this comparison, we used the microaveraged F-score measure. For each classifier, $\mathcal{G}_f$, we first compute its microaverage precision p and recall r by summing over all the individual decisions it made on the test set:

$$r(\mathcal{G}_f) = \frac{\sum_{k=1}^{K}\theta(k,\mathcal{G}_f)}{\sum_{k=1}^{K}(\theta(k,\mathcal{G}_f)+\psi(k,\mathcal{G}_f))},$$

$$p(\mathcal{G}_f) = \frac{\sum_{k=1}^{K}\theta(k,\mathcal{G}_f)}{\sum_{k=1}^{K}(\theta(k,\mathcal{G}_f)+\phi(k,\mathcal{G}_f))},$$

where $\theta(k,\mathcal{G}_f), \phi(k,\mathcal{G}_f)$ and $\psi(k,\mathcal{G}_f)$ respectively denote the true positive, false positive and false negative documents in class k found by $\mathcal{G}_f$. The F-score measure is then defined as:

$$F(\mathcal{G}_f) = \frac{2p(\mathcal{G}_f)r(\mathcal{G}_f)}{p(\mathcal{G}_f)+r(\mathcal{G}_f)}.$$

4.3. Results

We compared the performance of the models on the four datasets by varying the percentage of labeled examples in the training set and using *10*-fold cross validation (CV). We performed 10 runs for each of the folds and we calculated the average F-score (as we initialize some of the training hyper-parameters at random, we wanted to ensure that the results do not depend on this random initialization). In order to evaluate the significance of the observed differences in performance, we performed a t-test at the 5% significance level.

For the three benchmark datasets (`WebKB`, `Reuters` and `20Newsgroups`), we compared our semi-supervised PLSA models with two state-of-the-art methods in text classification: the semi-supervised Naive Bayes classifier (Nigam et al., 2000) and the transductive SVM classifier (SVM-light package Joachims, 1999). For the latter, we used a linear kernel and we optimized the cost parameter, using a nested cross-validation. We used the "one vs. all" strategy, choosing, for each example, the class with

[4] http://www.cs.cmu.edu/~webkb/

[5] http://ir.dcs.gla.ac.uk/resources/test_collections/cacm/

the maximum score. We could possibly obtain better results from TSVM with a non-linear kernel, but training would then become prohibitively expensive compared to the other models examined here. For all datasets (including XLS), we also compared the semi-supervised PLSA models with the fully supervised PLSA (Section 3.1).

In order to have an upper bound on the performance of the semi-supervised classifiers, we first compare the systems in a fully supervised way, that is when 100% of the documents in the training set have their true labels and are used for training the classifiers. Table 2 sums up these results. We include only one (supervised) PLSA model as there are no unlabeled documents, and therefore no "fake label" or "mislabeling error model".

Although the supervised PLSA yields significantly better performance on all three benchmark collections, all models reach roughly similar *F*-scores, except on the Reuters data, where the linear SVM is almost 5% below PLSA. This suggests that in a supervised settings, all three models offer reasonable and roughly comparable performance. We can then turn our attention to their behavior in a semi-supervised setting.

In our experiments, we train all models with varying amounts of annotation from 1% to 100%, using the rest of the training documents as unlabeled training data. In one CV fold of the 20Newsgroups collection, for example, this means that the ratio of labeled/unlabeled training data will vary from 144/14256 (1%) to 14400/0 (100%). In order to assess the gain in performance obtained from using the unlabeled data, we also trained a fully supervised PLSA model on the labeled portion of the training data.

Figs. 3–5 show the evolution of the test *F*-score on the three data collections (20Newsgroups, WebKB and Reuters) as the ratio of labeled to unlabeled documents increases. 5% in the *x*-axis means that 5% of the labeled documents ($|X_l|$) in the training sets were used for training, the 95% remaining being used as unlabeled training documents ($|X_u|$). The *ssPLSA-mems* uniformly outperforms the other models on these datasets. This is particularly clear for 20Newsgroups: With only 5% of labeled documents in the training set, the *F*-score of *ssPLSA-mems* is about 16 *F*-score points over that of *ssPLSA-fake* and 12 points over semi-supervised Naive Bayes. In addition, performance with only 5% of labeled data allows us to reach 90% of the maximum *F*-score obtained on 20Newsgroups, while labeling the remaining 95% documents allows to reach the last 10% of *F*-score. Note that the semi-supervised Naive Bayes model outperforms the semi-supervised PLSA using fake labels on the three benchmark datasets. This may be due to the fact that the "fake label" method puts the emphasis on better estimation of the confidence in the categorization decision rather than on raw performance improvement.

We also notice that on all benchmark data, the TSVM does clearly worse than all probabilistic models. In addition to the issue of the kernel choice, already mentioned above, we believe that this may be in part due to the fact that the TSVM model was initially designed for binary classification. The "one vs. all" strategy implemented to address these multiclass problems introduces one additional source of error.

Table 2
Performance of the models, fully supervised: Naive Bayes, PLSA and SVM classifier, on the 20Newsgroups, WebKB and Reuters collections. All models trained on 100% annotated data.

System	20Newsgroups *F*-score (%)	WebKB *F*-score (%)	Reuters *F*-score (%)
Naive Bayes	88.23	84.32	93.89
PLSA	$\|A\| = 40$	$\|A\| = 16$	$\|A\| = 14$
	89.72	**85.54**	**94.29**
SVM	88.98	85.15	89.50

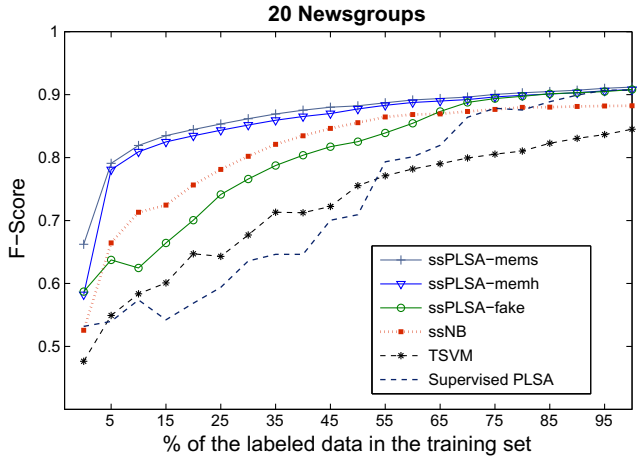

Fig. 3. *F*-score (*y*-axis) versus the percentage of labeled examples in the training set $|\mathcal{X}_l|/|\mathcal{X}|$ (*x*-axis), for various algorithms on the 20Newsgroups collection.

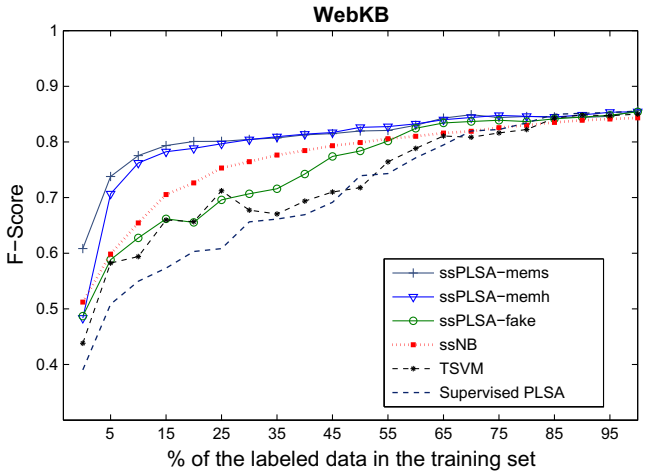

Fig. 4. *F*-score (*y*-axis) versus the percentage of labeled examples in the training set $|\mathcal{X}_l|/|\mathcal{X}|$ (*x*-axis), for various algorithms on the WebKB collection.

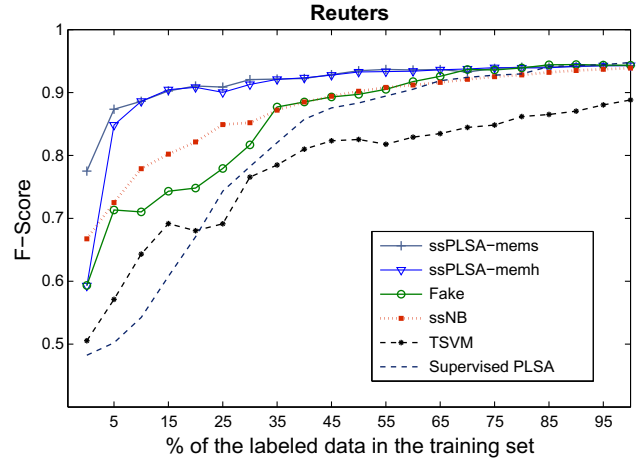

Fig. 5. *F*-score (*y*-axis) versus the percentage of labeled examples in the training set $|\mathcal{X}_l|/|\mathcal{X}|$ (*x*-axis), for various algorithms on the Reuters collection.

In order to empirically evaluate how the unlabeled documents improve the performance of the semi-supervised models, we compare them to a PLSA model trained in a fully supervised manner

Table 3
F-score for varying proportions of labeled-unlabeled training data, for the three variants of the semi-supervised PLSA (ssPLSA-fake, ssPLSA-memh, ssPLSA-mems) and different numbers of the latent topics $|A|$. Bold indicates statistically better results, measured using a *t*-test at the 5% significance level.

		1%	5%	20%	40%
20Newsgroups					
$\|A\| = 20$	ssPLSA-mems	65.96 ± 0.89	79.13 ± 0.11	83.59 ± 0.66	85.63 ± 0.42
	ssPLSA-memh	57.52 ± 0.59	76.42 ± 0.16	83.24 ± 0.57	85.54 ± 0.3
	ssPLSA-fake	57.04 ± 0.68	63.75 ± 0.78	70.05 ± 0.68	79.59 ± 0.28
$\|A\| = 40$	ssPLSA-mems	**66.23 ± 0.52**	**80.01 ± 0.23**	**84.42 ± 0.73**	**85.9 ± 0.85**
	ssPLSA-memh	58.24 ± 0.46	77.18 ± 0.2	83.47 ± 0.35	**85.76 ± 0.69**
	ssPLSA-fake	57.87 ± 0.41	59.75 ± 1.09	65.75 ± 0.98	78.86 ± 0.39
WebKB					
$\|A\| = 4$	ssPLSA-mems	60.25 ± 0.64	72.97 ± 0.24	79.84 ± 0.96	79.57 ± 0.26
	ssPLSA-memh	49.25 ± 0.73	70.03 ± 0.63	79.61 ± 0.52	79.52 ± 0.17
	ssPLSA-fake	44.42 ± 0.78	59.76 ± 0.84	68.94 ± 0.79	72.63 ± 0.59
$\|A\| = 16$	ssPLSA-mems	**60.56 ± 0.29**	**73.67 ± 0.33**	**80.56 ± 0.42**	**80.94 ± 0.72**
	ssPLSA-memh	49.84 ± 0.67	70.85 ± 0.51	80.67 ± 0.32	**80.83 ± 0.49**
	ssPLSA-fake	47.97 ± 0.87	62.76 ± 0.58	70.65 ± 0.86	73.78 ± 0.34

using only the fraction of labeled documents in the training set (Figs. 3–5). We can see that semi-supervised algorithms are able to take advantage of unlabeled data. For example, for the 20Newsgroups dataset (Fig. 3), with 5% labeled data, the fully supervised PLSA reaches a *F*-score of 52.5%, while *ssPLSA-fake* yields 63% and *ssPLSA-memh* and *ssPLSA-mems* achieve 79%. As we can notice on Figs. 4 and 5, the gain in performance obtained from the use of the unlabeled data is similar for the other two datasets (WebKB and Reuters).

4.3.1. Influence of the number of latent aspects

One interesting aspect of our experimental results is that the behavior of the ssPLSA variants is very different depending on the number of latent variables per class. This is illustrated in Table 3[6]. For the "fake label" approach, the behavior is different on both benchmarks. On 20Newsgroups, the performance tends to decrease when more components are added to the model, and the variability of the results increases. On WebKB, the performance consistently increases with more components (between +1.1% and +3.5% in *F*-score). As we saw previously, *ssPLSA-fake* consistently yields lower performance than the use of the mislabeling error model. In addition, *ssPLSA-memh* and *ssPLSA-mems* seem less sensitive to variation in the numbers of components. Table 3 shows that when the number of components per class doubles (from $|A| = 20$ to $|A| = 40$ for 20Newsgroups) or quadruples (from $|A| = 4$ to $|A| = 16$ for WebKB), the performance of both mislabeling approaches increases slightly, but consistently. In addition, the variability of the results is mostly well contained and typically smaller than for *ssPLSA-fake*.

These results confirm that *ssPLSA-mems* provides the best performance, especially when little annotation is available, as shown in the 1% column. In addition, the mislabeling error model seems to be relatively immune to a moderate amount of misspecification in the model structure.

4.3.2. Performance on XLS data

In addition to the three benchmark collections used above, we also tested our semi-supervised models on data coming from an actual use case. For the XLS dataset, we compare the semi-supervised PLSA models only with the supervised version of PLSA. This restriction stems from business constraints, as the current technology used and developed by Xerox relies on PLSA and a complete paradigm change to, for example, SVM-based methods would not be considered. In addition, the more extensive comparison performed on the benchmark suggests that the best semi-supervised PLSA model consistently outperforms both Naive Bayes and TSVM.

Fig. 6 shows how the *F*-score performance evolves with an increasing ratio of labeled to unlabeled examples. Again, the semi-supervised PLSA using a mislabeling error model outperforms other models, especially when very few labels are available. Above 10% of labeled examples, the performance of all classifiers does not change much. Although the gains in performance are more limited on this dataset (due to noise in document contents and training labels, as explained before), these results support the conclusions we

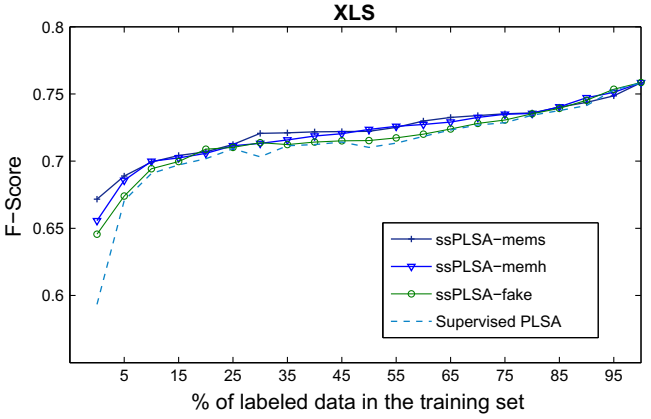

Fig. 6. *F*-score (*y*-axis) versus the percentage of labeled examples in the training set $|\mathcal{X}_l|/|\mathcal{X}|$ (*x*-axis), for various algorithms on the XLS collection.

Table 4
Comparison of the two mislabeling error model variants of ssPLSA, *ssPLSA-memh* and *ssPLSA-mems*, when the ratio of labeled-to-unlabeled data is very low (0.3% to 1%).

Ratio	Algorithm	20Newsgroups	WebKB	Reuters	XLS
		F-score	*F*-score	*F*-score	*F*-score
0.3%	ssPLSA-memh	32.62	38.82	47.76	61.41
	ssPLSA-mems	**44.05**	**48.78**	**66.34**	**65.16**
0.5%	ssPLSA-memh	41.26	40.86	52.02	64.52
	ssPLSA-mems	**52.46**	**51.55**	**68.74**	**66.19**
0.8%	ssPLSA-memh	51.2	44.16	57.42	64.87
	ssPLSA-mems	**60.62**	**56.33**	**75.11**	**67.04**
1%	ssPLSA-memh	58.24	49.84	66.93	65.57
	ssPLSA-mems	**66.23**	**60.56**	**77.53**	**67.17**

[6] For the 20Newsgroups dataset, the difference in performance w.r.t. figures reported in (Krithara et al., 2008) is due to the fact that here we average over 10 runs of CV, instead of 10 random labeled/unlabeled/test splits. Note that the performance reported here is within the error bars mentioned in (Krithara et al., 2008).

reached on the benchmark collections: semi-supervised learning applied to the PLSA aspect model does actually help reach good performance faster, and *ssPLSA-mems* outperforms all other models, especially for low annotation ratio, a property we will now investigate further.

4.3.3. Low annotation behaviour

From Figs. 3–6, we can see that the two variants of the mislabeling error model, *ssPLSA-memh* (hard cluster to class assignment) and *ssPLSA-mems* (soft assignment), behave similarly for most of the range of labeling proportion. The difference lies mainly in the region where the proportion of labeled documents is lowest, where *ssPLSA-mems* seems to perform better. We investigate this effect further in Table 4, where we report the *F*-score of both variants when the proportion of labeled example goes below 1%, on all four collections (the three benchmarks and `XLS`).

Table 4 shows that the performance of *ssPLSA-mems* is consistently better than that of *ssPLSA-memh*, by a large margin. On the benchmark collections, the *F*-score for the soft assignment strategy is 8% to 18% higher than for the hard assignment strategy. For the Xerox data, the difference is more modest (1.6% to 3.7%), but still consistently in favour of *ssPLSA-mems*. This suggest that the soft assignment variant should be the model of choice when little data can be annotated.

5. Conclusion

In this work, a variant of the semi-supervised PLSA algorithm has been presented. A mislabeling error model is incorporated in the generative aspect model. Experiments on the `20Newsgroups`, `WebKB`, and `Reuters` datasets validate a consistent significant increase in performance. These promising results suggest that good classification accuracy can be achieved with a small number of labeled documents while keeping most of the training set unlabeled. They also suggest that the proposed *ssPLSA-mem* is particularly effective when the ratio of annotated data is very low, and is also relatively immune to some mis-specification of the model structure. The performance will therefore not suffer too much in the usual situation where the correct number of latent aspects is unknown and must be approximated. Another advantage of the semi-supervised PLSA model introduced here is that it can be used directly to perform multiclass classification tasks, and as a result, it is easily applicable to real world problems.

A next step would be to try to combine the "fake label" approach as presented in (Krithara et al., 2006; Gaussier and Goutte, 2005), with the mislabeling error model. This combination would benefit from the advantages of each of the two versions and it would hopefully improve the performance of our classifier. Also, a combination with active learning could be tried out using different active learning methods.

Acknowledgment

This work was supported in part by the IST Program of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

References

Ambroise, C., Govaert, G., 2000. EM for partially known labels. In: Proceedings of the Seventh International Federation of Classification Societies. Namur, Belgium, pp. 161–166.

Amini, M., Gallinari, P., 2003. The use of unlabeled data to improve supervised learning for text summarization. In: SIGIR. pp. 105–112.

Amini, M., Gallinari, P., 2003. Semi-supervised learning with explicit misclassification modeling. In: Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI). pp. 555–560.

Amini, M., Laviolette, F., Usunier, N., 2009. A transductive bound for the voted classifier with an application to semi-supervised learning. In: Advances in Neural Information Processing Systems. 21. pp. 65–72.

Blei, D., Ng, A., Jordan, M., 2003. Latent Dirichlet allocation. J. Machine Learn. Res. 3.

Blum, A., Mitchell, T., 1998. Combining labeled and unlabeled data with co-training. In: Proceedings of the Workshop on Computational Learning Theory. Morgan Kaufmann Publishers, pp. 92–100.

Castelli, V., Cover, T., 1995. On the exponential value of labeled samples. Pattern Recognition Lett. 16 (1), 105–111.

Castelli, V., Cover, T., 1996. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. IEEE Trans. Inf. Theory 42 (6), 2102–2117.

Chhikara, R., McKeon, J., 1984. Linear discriminant analysis with misallocation in training samples. J. Am. Stat. Assoc. 79, 899–906.

Chittineni, C.B., 1980. Learning with imperfectly labeled patterns. Pattern Recognit. 12 (5), 281–291.

Dempster, A., Laird, N., Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc., Ser. B 39 (1).

Fralick, S.C., 1967. Learning to recognize patterns without a teacher. IEEE Trans. Inf. Theory 13 (1), 57–64.

Gaussier, E., Goutte, C., 2005. Learning from partially labelled data-with confidence. In: Proceedings of Learning with Partially Classified Training Data – ICML'05 workshop.

Gaussier, E., Goutte, C., Popat, K., Chen, F., 2002. A hierarchical model for clustering and categorising documents. In: Advances in Information Retrieval. Springer, pp. 229–247.

Grandvalet, Y., Bengio, Y., 2005. Semi-supervised learning by entropy minimization. Advances in Neural Information Processing Systems, 17. MIT Press, Cambridge, MA, pp. 529–536.

Hofmann, T., 2001. Unsupervised learning by probabilistic latent semantic analysis. Machine Learn. 42 (1–2), 177–196.

Joachims, T., 1999. Transductive inference for text classification using support vector machines. In: Proceedings of 16th International Conference on Machine Learning (ICML). Bled, Slovenia, pp. 200–209.

Krishnan, T., 1988. Efficiency of learning with imperfect supervision. Pattern Recognit. 21 (2), 183–188.

Krishnan, T., Nandy, S.C., 1987. Discriminant analysis with a stochastic supervisor. Pattern Recognit. 20 (4), 379–384.

Krithara, A., Amini, M.-R., Renders, J.-M., Goutte, C., 2008. Semi-supervised document classification with a mislabeling error model. In: European Conference on Information Retrieval (ECIR). Glasgow, Scotland, pp. 370–381.

Krithara, A., Goutte, C., Renders, J., Amini, M., 2006. Reducing the annotation burden in text classification. In: Proceedings of the First International Conference on Multidisciplinary Information Sciences and Technologies (InSciT 2006). Merida, Spain.

Lachenbrunch, P., 1974. Discriminant analysis when the initial samples are misclassified. II: Non-random misclassification models. Technometrics 16, 419–424.

Lawrence, N., Schlkopf, B., 2001. Estimating a kernel Fisher discriminant in the presence of label noise. In: Proceedings of the 18th International Conference on Machine Learning (ICML). San Francisco, CA, USA, pp. 306–313.

McLachlan, G., 1972. Asymptotic results for discriminant analysis when the initial samples are misclassified. Technometrics 14, 415–422.

McLachlan, G.L., 1992. Discriminant Analysis and Statistical Pattern Recognition. Wiley, New York, N.Y.

Miller, D.J., Uyar, H.S., 1997. A mixture of experts classifier with learning based on both labelled and unlabelled data. In: Proceedings of NIPS. pp. 571–577.

Minka, T., Lafferty, J., 2002. Expectation-propagation for the generative aspect model. In: Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence. pp. 352–359.

Ng, A., Jordan, M., 2002. On discriminative vs. generative classifiers. A comparison of logistic regression and naive bayes. In Advances in Neural Information Processing Systems, Vol. 14. MIT Press, Cambridge, MA, pp. 841–848.

Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.M., 2000. Text classification from labeled and unlabeled documents using EM. Machine Learn. 39 (2/3), 103–134.

O'Neill, T., 1978. Normal discrimination with unclassified observations. Am. Stat. Assoc. 73 (36), 821–826.

Saul, L., Pereira, F., 1997. Aggregate and mixed-order Markov models for statistical language processing. In: Proceedings of the Second Conference on Empirical Methods in Natural Language Processing. Somerset, New Jersey, pp. 81–89.

Titterington, D., 1989. An alternative stochastic supervisor in discriminant analysis. Pattern Recognit. 22 (1), 91–95.

Vapnik, V., 1998. Statistical Learning Theory. John Wiley and Sons, New York.

Vittaut, J.-N., Amini, M.-R., Gallinari, P., 2002. Learning classification with both labeled and unlabeled data. In: Proceedings of the 13th European Conference on Machine Learning. pp. 468–476.