Journal of Intelligent Information Systems, 18:2/3, 195–217, 2002

Hidden Markov Models for Text Categorization in Multi-Page Documents

PAOLO FRASCONI paolo@dsi.unifi.it
GIOVANNI SODA giovanni@dsi.unifi.it
ALESSANDRO VULLO vullo@dsi.unifi.it
Department of Systems and Computer Science, University of Florence, Firenze I-50139, Italy

Abstract. In the traditional setting, text categorization is formulated as a concept learning problem where each instance is a single isolated document. However, this perspective is not appropriate in the case of many digital libraries that offer as contents scanned and optically read books or magazines. In this paper, we propose a more general formulation of text categorization, allowing documents to be organized as *sequences* of pages. We introduce a novel hybrid system specifically designed for multi-page text documents. The architecture relies on hidden Markov models whose emissions are bag-of-words resulting from a multinomial word event model, as in the generative portion of the Naive Bayes classifier. The rationale behind our proposal is that taking into account contextual information provided by the whole page sequence can help disambiguation and improves single page classification accuracy. Our results on two datasets of scanned journals from the Making of America collection confirm the importance of using whole page sequences. The empirical evaluation indicates that the error rate (as obtained by running the Naive Bayes classifier on isolated pages) can be significantly reduced if contextual information is incorporated.

Keywords: text categorization, multi-page documents, hidden Markov models, Naive Bayes, digital libraries

1. Introduction

The recent explosion of online textual information has significantly increased the demand for intelligent agents capable of performing tasks such as personalized information filtering, semantic document indexing, information extraction, and automatic metadata generation. Although a complete answer may require in-depth approaches involving full understanding of natural language, text categorization is a simpler but effective technique that can contribute to the solution of the above problems.

Originally posed as a problem in information retrieval, text categorization can be conveniently formulated as a supervised learning problem. In this setting, a machine learning algorithm takes as input a set of labeled example documents (where the label indicates which category the example belongs to) and attempts to infer a function that will map new documents into their categories. Several algorithms have been proposed within this framework, including regression models (Yang and Chute, 1994), inductive logic programming (Cohen, 1995), probabilistic classifiers (Koller and Sahami, 1997; Lewis and Gale, 1994; Mitchell, 1997), decision trees (Lewis and Ringuette, 1994), neural networks (Ng et al., 1997), and more recently support vector machines (Joachims, 1998).

Research on text categorization has been mainly focused on non-structured documents. In the typical approach, inherited from information retrieval, each document is represented by a sequence of words, and the sequence itself is normally flattened down to a simplified representation called *bag-of-words*. This is like representing each document as a feature-vector, where features are words in the vocabulary and components of the feature-vector are statistics such as word counts in the document. Although such a simplified representation is appropriate for relatively flat documents (such as email and news messages), other types of documents are internally structured and this structure should be exploited in the representation to better inform the learner.

In this paper we are interested in the domain of digital libraries and, in particular, collections of digitized books or magazines, with text extracted by an Optical Character Recognition (OCR) system. Unlike email or news documents, books and magazines are *multi-page* documents and the simplest level of structure that can be exploited is the serial order relation defined among single pages. In these domains, the solution to problems such as automatic metadata extraction can be helped by a classifier that assigns a category to each page of the document. The task we consider is the automatic categorization of each page according to its (semantic) contents.[1]

Exploiting the serial order relation among pages within a single document can be expected to improve classification accuracy when compared to a strategy that simply classifies each page separately. This is because sequences of pages in documents such as books or magazines often follow regularities such as those implied by typographical and editorial conventions. Consider for example the domain of books and suppose categories of interest include `title-page`, `dedication-page`, `preface-page`, `index-page`, `table-of-contents`, `regular-page`, and so on. Even in this very simple case we can expect constraints about the valid sequences of page categories in a book. For example, `title-page` is very unlikely to follow `index-page` and, similarly, `dedication-page` is more likely to follow `title-page` than `preface-page`. Constraint of this type can be captured and modeled using a stochastic grammar (see, e.g., figure 4 later on). Thus, information about the category of a given page can be gathered not only by examining the contents of that page, but also by examining the contents of other pages in the sequence. Since contextual information can significantly help to disambiguate between page categories, we expect classification accuracy to improve if the learner has access to whole sequences instead of single-page documents.

In this paper we combine several algorithmic ideas to solve the problem of text categorization in the domain of multi-page documents. First, we use an algorithm similar to those described in Stolcke and Omohundro (1993) and McCallum et al. (2000) for inducing a stochastic regular grammar over sequences of page categories. Second, we introduce a hidden Markov model (HMM) that can deal with sequences of bag-of-words. Each state in the HMM is associated with a unique page category. Emissions are modeled by a multinomial distribution over word events, like in the generative component of the Naive Bayes classifier. The HMM is trained from (partially) labeled page sequences, i.e. state variables are partially observed in the training set. Unobserved states (which is the common setting in most classic applications of HMMs) arise here when document pages are partially unlabeled, like in the framework described in Joachims (1999) and Nigam et al.

(2000). Finally, we solve the categorization problem by running the Viterbi algorithm on the trained HMM. For each new (unseen) document, this algorithm outputs a sequence of page categories having maximum posterior[2] probability. The method is somewhat related to recent applications of HMMs to information extraction (Freitag and McCallum, 2000; McCallum et al., 2000) but the output labeling in our case is associated with the entire stream of text contained into a page, while in Freitag and McCallum (2000) and McCallum et al. (2000) the HMM is used to attach labels to single words of shorter portions of text.

Our approach is validated on two real datasets consisting of 95 issues of the *American Missionary*, and 54 issues of the *Scribners Monthly*, two journals included in the "Making of America" collection (Shaw and Blumson, 1997). In spite of text noise due to optical character recognition, our system achieves good page classification accuracy. More importantly, we show that incorporating contextual information significantly reduces classification error, both in the case of completely labeled example documents, and when unlabeled documents are included in the training set.

2. Background

Let d be a generic multi-page document, and let d_t denote the t-th page within the document. The standard categorization task consists of learning from examples a function $f : d_t \rightarrow \{c^1, \ldots, c^K\}$ that maps each page d_t into one out of K classes.

2.1. The Naive Bayes classifier

The above task can also be reformulated in probabilistic terms as the estimation of the conditional probability $P(C_t = c^k \mid d_t)$, being C_t a multinomial class variable. In so doing, f can be computed using Bayes' decision rule, i.e. $f(d)$ is the class with higher posterior probability $P(C_t = c^k \mid d_t) \propto P(d_t \mid C_t = c^k)P(C_t = c^k)$. The model is characterized by the so-called Naive Bayes assumption, prescribing that word events (each occurrence of a given word in the page corresponds to one event) are conditionally independent *given* the page category. As a result, the class conditional probabilities can be factorized as

$$P\left(d_t \mid C_t = c^k\right) \propto \prod_{i=1}^{|d_t|} P\left(w_t^i \,\middle|\, C_t = c^k\right) \tag{1}$$

where $|d_t|$ denotes the length of page d_t and w_t^i is the i-th word in the page. This conditional independence assumption is graphically represented by the Bayesian network[3] shown in figure 1.

Although the basic assumption is clearly false in the real world, the model works well in practice since classification requires finding a good discriminating function, not necessarily a very accurate model of the involved probability distributions. Training consists of estimating model's parameter from a dataset $\mathcal{D}$ of labeled documents (see, e.g. Mitchell, 1997).

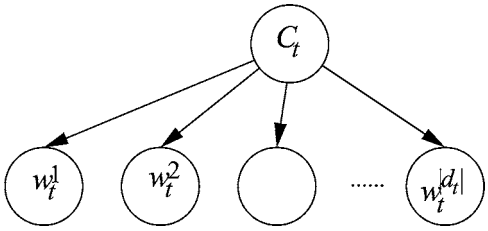

Figure 1. Bayesian network for the Naive Bayes classifier.

2.2. *Hidden Markov models*

HMMs have been introduced several years ago as a tool for probabilistic sequence modeling. The interest in this area developed particularly in the Seventies, within the speech recognition research community (Rabiner, 1989). During the last years a large number of variants and improvements over the standard HMM have been proposed and applied. Undoubtedly, Markovian models are now regarded as one of the most significant state-of-the-art approaches for sequence learning. Besides several applications in pattern recognition and molecular biology, HMMs have been also applied to text related tasks, including natural language modeling (Charniak, 1993) and, more recently, information retrieval and extraction (Freitag and McCallum, 2000; McCallum et al., 2000). The recent view of the HMM as a particular case of Bayesian networks (Bengio and Frasconi, 1995; Lucke, 1995; Smyth et al., 1997) has helped their theoretical understanding and the ability to conceive extensions to the standard model in a sound and formally elegant framework.

An HMM describes two related discrete-time stochastic processes. The first process pertains to hidden discrete state variables, denoted X_t, forming a first-order Markov chain and taking realizations on a finite set $\{x^1, \ldots, x^N\}$. The second process pertains to observed variables or *emissions*, denoted D_t. Starting from a given state at time 0 (or given an initial state distribution $P(X_0)$) the model probabilistically transitions to a new state X_1 and correspondingly emits observation D_1. The process is repeated recursively until an end state is reached. Note that, as this form of computation may suggest, HMMs are closely related to stochastic regular grammars (Charniak, 1993). The Markov property prescribes that X_{t+1} is conditionally independent of $X_1, \ldots, X_{t-1}$ given X_t. Furthermore, it is assumed that D_t is independent of the rest given X_t. These two conditional independence assumptions are graphically depicted using the Bayesian network of figure 2. As a result, an HMM is fully specified by the following conditional probability distributions:[4]

$$\begin{array}{ll} P(X_t \mid X_{t-1}) & \text{(transition distribution)} \\ P(D_t \mid X_t) & \text{(emission distribution)} \end{array} \tag{2}$$

Since the process is stationary, the transition distribution can be represented as a square stochastic matrix whose entries are the transition probabilities $P(X_t = x^i \mid X_{t-1} = x^j)$, abbreviated as $P(x^i \mid x^j)$ in the following. In the classic literature, emissions are restricted

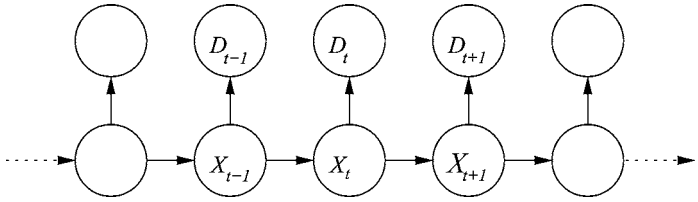

Figure 2. Bayesian networks for standard HMMs.

to be symbols in a finite alphabet or multivariate continuous variables (Rabiner, 1989). As explained in the next section, our model allows emissions to be bag-of-words.

3. The multi-page classifier

We now turn to the description of our classifier for multi-page documents. In this case the categorization task consists of learning from examples a function that maps the whole document sequence $d_1, \ldots, d_T$ into a corresponding sequence of page categories, $c_1, \ldots, c_T$. This section presents the architecture and the asociated algorithms for grammar extraction, training, and classification.

3.1. Architecture

The system is based on an HMM whose emissions are associated with entire pages of the document. Thus, the realizations of the observation D_t are bag-of-words representing the text in the t-th page of the document. HMM states are related to pages categories by a deterministic function ϕ that maps state realizations into page categories. We assume that ϕ is a surjection but not a bijection, i.e. that there are more state realizations than categories. This enriches the expressive power of the model, allowing different transition behaviors for pages of the same class, depending on where the page is actually encountered within the sequence. However, if the page *contents* depends on the category but not on the context of the category within the sequence,[5] multiple states may introduce too many parameters and it may be convenient to assume that

$$P(D_t \mid x^i) = P(D_t \mid x^j) = P(D_t \mid c^k) \text{ if } \phi(x^i) = \phi(x^j) = c^k. \tag{3}$$

This constrains emission parameters to be the same for a given page category, a form of parameters sharing that may help to reduce overfitting. The emission distribution is modeled by assuming conditional word independence given the class, like in Eq. (1):

$$P(d_t \mid c^k) = \prod_{i=1}^{|d_t|} P\left(w_t^i \mid c^k\right). \tag{4}$$

Therefore, the architecture can be graphically described as the merging of the Bayesian networks for HMMs and Naive Bayes, as shown in figure 3. We remark that the state (and

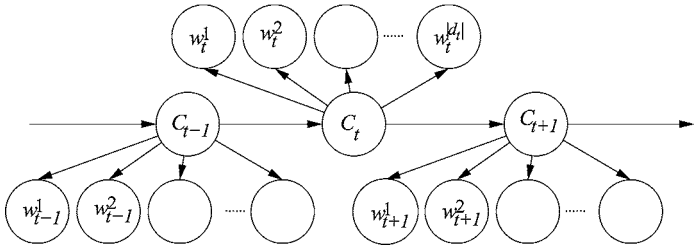

Figure 3. Bayesian network describing the architecture of the sequential classifier.

hence the category) at page t depends not only on the contents of that page, but also on the contents of all other pages in the document, summarized into the HMM states. This probabilistic dependency implements the mechanism for taking contextual information into account.

The algorithms used in this paper are derived from the literature on Markov models (Rabiner, 1989), inference and learning in Bayesian networks (Pearl, 1988; Heckerman, 1997; Jensen, 1996) and classification with Naive Bayes (Lewis and Gale, 1994; Kalt, 1996). In the following we give details about the integration of all these methods.

3.2. *Induction of HMM topology*

The *structure* or topology of an HMM is a representation of the allowable transitions between hidden states. More precisely, the topology is described by a directed graph whose vertices are state realizations $\{x^1, \ldots, x^N\}$, and whose edges are the pairs (x^j, x^i) such that $P(x^i \mid x^j) \neq 0$. An HMM is said to be *ergodic* if its transition graph is fully-connected. However, in almost all interesting application domains, less connected structures are better suited for capturing the observed properties of the sequences being modeled, since they convey domain prior knowledge. Thus, starting from the right structure is an important problem in practical Hidden Markov modeling. As an example, consider figure 4, showing a (very simplified) graph that describes transitions between the parts of a hypothetical set of books. Possible state realizations are {start, title, dedication, preface, toc, regular, index, end} (note that in this simplified example ϕ is a one-to-one mapping).

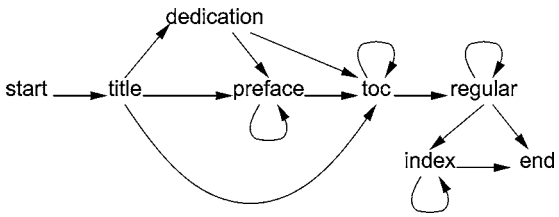

Figure 4. Example of HMM transition graph.

While a structure of this kind could be hand-crafted by a domain expert, it may be more advantageous to learn it automatically from data. We now briefly describe the solution adopted to automatically infer HMM transition graphs from sample multi-page documents. Let us assume that all the pages of the available training documents are labeled with the class they belong to. One can then imagine to take advantage of the observed labels to search for an effective structure in the space of HMMs topologies. Our approach is based on the application of an algorithm for data-driven model induction adapted from previous works on construction of HMMs of text phrases for information extraction (McCallum et al., 2000). The algorithm starts by building a structure that can only "explain" the available training sequences (a maximally specific model). This initial structure has as many paths (from the initial to the final state) as there are training sequences. Every path is associated with one sequence of pages, i.e. a distinct state is created for every page in the training set. Each state x is labeled by $\phi(x)$, the category of the corresponding page in the document. Note that, unlike the example shown in figure 4, several states are generated for the same category. The algorithm then iteratively applies merging heuristics that collapse states so as to augment generalization capabilities over unseen sequences. The first heuristic, called neighbor-merging, collapses two states x and x' if they are neighbors in the graph and $\phi(x) = \phi(x')$. The second heuristic, called V-merging, collapses two states x and x' if $\phi(x) = \phi(x')$ and they share a transition from or to a common state, thus reducing the branching factor of the structure.

3.3. Inference and learning

Given the HMM topology extracted by the algorithm described above, the learning problem consists of determining transition and emission parameters. One important distinction that needs to be made when training Bayesian networks is whether or not all the variables are observed. Assuming complete data (all variables observed), maximum likelihood estimation of the parameters could be solved using a one-step algorithm that collects sufficient statistics for each parameter (Heckerman, 1997). In our case, data are complete if and only if the following two conditions are met:

1. there is a one-to-one mapping between HMM states and page categories (i.e. $N = K$ and for $k = 1, \ldots, N$, $\phi(x^k) = c^k$), and
2. the category is known for each page in the training documents, i.e. the dataset consists of sequences of pairs $(\{d_1, c_1^*\}, \ldots, \{d_T, c_T^*\})$, being c_t^* the (known) category of page t and being T the number of pages in the document.

Under these assumptions, estimation of transition parameters is straightforward and can be accomplished as follows:

$$P(x^i \mid x^j) = \frac{N(c^i, c^j)}{\sum_{\ell=1}^{N} N(c^\ell, c^j)} \tag{5}$$

where $N(c^i, c^j)$ is the number of times a page of class c^i follows a page of class c^j in the training set. Similarly, estimation of emission parameters in this case would be accomplished

exactly like in the case of the Naive Bayes classifier (see, e.g. Mitchell (1997)):

$$P(w^\ell \mid c^k) = \frac{1 + N(w^\ell, c^k)}{|V| + \sum_{j=1}^{|V|} N(w^j, c^k)} \tag{6}$$

where $N(w^\ell, c^k)$ is the number of occurrences of word w^ℓ in pages of class c^k and $|V|$ is the vocabulary size ($1/|V|$ corresponds to a Dirichlet prior over the parameters (Heckerman, 1997) and plays a regularization role for whose words which are very rare within a class).

Conditions 1 and 2 above, however, are normally not satisfied. First, in order to model more accurately different contexts in which a category may occur, it may be convenient to have multiple distinct HMM states for the same page category. This implies that page labels do not determine a unique state path. Second, labeling pages in the training set is a time consuming process that needs to be performed by hand and it may be important to use also unlabeled documents for training (Joachims, 1999; Nigam et al., 2000). This means that label c_t^* may be not available for some t. If assumption 2 is satisfied but assumption 1 is not, we can derive the following approximated estimation formula for transition parameters:

$$P(x^i \mid x^j) = \frac{N(x^i, x^j)}{\sum_{\ell=1}^{N} N(x^\ell, x^j)} \tag{7}$$

where $N(x^i, x^j)$ counts how many times state x^i follows x^j during the state merge procedure described in Section 3.2. However, in general, the presence of hidden variables requires an *iterative* maximum likelihood estimation algorithm, such as gradient ascent or expectation-maximization (EM). Our implementation uses the EM algorithm, originally formulated in Dempster et al. (1977) and usable for any Bayesian network with local conditional probability distributions belonging to the exponential family (Heckerman, 1997). Here the EM algorithm essentially reduces to the Baum-Welch form (Rabiner, 1989) with the only modification that some evidence is entered into state variables. Since multiple states are associated with a category and even for labeled documents only the page category is known, state evidence takes the form of *findings* (Jensen, 1996). State evidence is taken into account in the E-step by changing forward propagation as follows:

$$\alpha_t(j) = \begin{cases} 0 & \text{if } \phi(x^j) \neq c_t^* \\ \displaystyle\sum_{i=1}^{N} \alpha_{t-1}(i) P(x^j \mid x^i) P(d_t \mid x^j) & \text{otherwise} \end{cases} \tag{8}$$

where

$$\alpha_t(i) \doteq P(d_1, \ldots, d_t, X_t = x^i)$$

is the forward variable in the Baum-Welch algorithm. The emission probability $P(d_t|x^j)$ is obtained from Eq. (4), using $c^k = \phi(x^j)$.

The M-step is performed in the standard way for transition parameters, by replacing counts in Eq. (5) with their expectations given all the observed variables. Emission probabilities

are also estimated using expected word counts. If parameters are shared as indicated in Eq. (3), these counts should be summed over states having the same label. Thus, in the case of incomplete data, Eq. (6) is replaced by

$$P(w^\ell \mid c^k) = \frac{S + \sum_{s=1}^{S} \sum_{t=1}^{T} N(w^\ell, c^k) \sum_{i:\phi(x^i)=c^k} P(X_t = x^i \mid d_1, \ldots, d_T)}{S \cdot |V| + \sum_{j=1}^{|V|} \sum_{s=1}^{S} \sum_{t=1}^{T} N(w^j, c^k) \sum_{i:\phi(x^i)=c^k} P(X_t = x^i \mid d_1, \ldots, d_T)} \tag{9}$$

where S is the number of training sequences, $N(w^\ell, c^k)$ is the number of occurrences of word w^ℓ in pages of class c^k and $P(X_t = x^i \mid d_1, \ldots, d_T)$ is the probability of being in state x^i at page t given the observed sequence of pages $d_1 \ldots d_T$. Readers familiar with HMMs should recognize that the latter quantity can be computed by the Baum-Welch procedure during the E-step. The sum on p extends over training sequences, while the sum on t extends over pages of the p-th document in the training set. The E- and M-steps are iterated until a local maximum of the (incomplete) data likelihood is reached.

Note that if page categories are observed, it is convenient to use the estimates computed with Eq. (7) as a starting point, rather than using random initial parameters. Similarly, an initial estimate of the emission parameters can be obtained from Eq. (6).

It is interesting to point out a related application of the EM algorithm for learning from labeled and unlabeled documents (Nigam et al., 2000). In that paper, the only concern was to allow the learner to take advantage of unlabeled documents in the training set. As a major difference, the method in Nigam et al. (2000) assumes flat single-page documents and, if applied to multi-page documents, would be equivalent to a zero-order Markov model that cannot take contextual information into account.

3.4. Page classification

Given a document of T pages, classification is performed by first computing the sequence of states $\hat{x}_1, \ldots, \hat{x}_T$ that was most likely to have generated the observed sequence of pages, and then mapping each state to the corresponding category $\phi(\hat{x}_t)$. The most likely state sequence can be obtained by running an adapted version of Viterbi's algorithm, whose more general form is the max-propagation algorithm for Bayesian networks described in Jensen (1996). Briefly, the following quantity

$$\delta_t^j \doteq \max_{x_1,\ldots,x_{t-1}} P(X_1, \ldots, X_t = x^j, d_1, \ldots, d_t) \tag{10}$$

is computed using the following recursion:

$$\delta_1^j = P(X_1 = x^j) P(d_1 \mid x^j) \tag{11}$$

$$\delta_t^j = \left[\max_{i=1,\ldots,N} \delta_{t-1}^i P(x^j \mid x^i) \right] P(d_t \mid x^j) \tag{12}$$

$$\psi_t^j = \arg \max_{i=1,\ldots,N} \delta_{t-1}^i P(x^j \mid x^i). \tag{13}$$

The optimal state sequence is then retrieved by backtracking:

$$\hat{x}_T = \arg \max_{i=1,\ldots,N} \delta_T^i, \tag{14}$$

$$\hat{x}_t = \psi_t^{\hat{x}_{t+1}}. \tag{15}$$

Finally, categories are obtained as $\hat{c}_t = \phi(\hat{x}_t)$. By contrast, note that the Naive Bayes classifier would compute the most likely categories as

$$\hat{c}_t = \arg \max_{j=1,\ldots,K} P(c^j) P(d_t \mid c^j). \tag{16}$$

Comparing Eqs. (11)–(15) to Eq. (16) we see that both classifiers rely on the same emission model $P(d_t \mid c^j)$ but while Naive Bayes employs the prior class probability to compute its final prediction, the HMM classifier takes advantage of a dynamic term (in square brackets in Eq. (12)) that incorporates grammatical constraints.

4. Experimental results

In this section, we describe a set of experiments that give empirical evidence of the effectiveness of the proposed model. The main purpose of our experiments was to make a comparison between our multi-page classification approach and a traditional isolated page classification system, like the well known Naive Bayes text classifier. The evaluation has been conducted over real-world documents that are naturally organized in the form of page sequences. We used two different datasets associated with two journals in the Making of America (MOA) collection. MOA is a joined project between the University of Michigan and Cornell University (see `http://moa.umdl.umich.edu/about.html` and Shaw and Blumson (1997)) for collecting and making available digitized information about history and evolution processes of the American society between the XIX and the XX century.

4.1. Datasets

The first dataset is a subset of the journal *American Missionary*, a sociological magazine with strong Christian guidelines. The task consists of correctly classifying pages of previously unseen documents into one of the ten categories described in Table 1. Most of these categories are related to the topic of the articles, but some are related to the parts of the journal (i.e. Contents, Receipts, and Advertisements). The dataset we selected contains 95 issues from 1884 to 1893, for a total of 3222 OCR text pages. Special issues and final report issues (typically November and December issues) have been removed from the dataset as they contain categories not found in the rest. The ten categories are temporally stable over the 1883–1893 time period.

The second dataset is a subset of Scribners Monthly, a recreational and cultural magazine printed in the second half of the XIX century. Table 2 describes the categories we have selected for this classification task. The filtered dataset contains a total of 6035 OCR text pages, organized into issues ranging from year 1870 to 1875. Although spanning a shorter temporal interval, the number of pages in this second dataset is larger than in the first one because issues are about 3–4 times longer.

Table 1. Categories in the *American Missionary* domain.

Name	Description
1. Contents	Cover and index of surveys
2. Editorial	Editorial articles
3. The South	Afro-Americans' surveys
4. The Indians	American Indians' surveys
5. The Chinese	Reports from China missions
6. Bureau of Women's work	Articles about female condition
7. Children's Page	Education and childhood
8. Communications	Magazine information
9. Receipts	Lists of founders
10. Advertisements	Contents is mostly graphic, with little text description

Table 2. Categories in the *Scribners Monthly* domain.

Name	Description
1. Article	Generic articles
2. Books and Authors at Home and Abroad	Book reviews
3. Contents	Table of contents
4. Culture and Progress	Broad cultural news
5. Etchings	Poems or tales
6. Home and Society	Articles on home living
7. Nature and Science	Scientific articles
8. The Old Cabinet	Articles on fine arts
9. Topics of the Time	News reports

Category labels for the two datasets were obtained semi-automatically, starting from the MOA XML files supplied with the documents collections. The assigned categories were then manually checked. In the case of a page containing the end and the beginning of two articles belonging to different categories, the page was assigned the category of the ending article.

Each page within a document is represented as a bag-of-words, counting the number of word occurrences within the page. It is worth remarking that in both datasets, instances are text documents output by an OCR system. Imperfections of the recognition algorithm and the presence of images in some pages yields noisy text, containing misspelled or nonexistent words, and trash characters (see Bicknese (1998) for a report of OCR accuracy in the MOA digital library). Although these errors may negatively affect the learning process and subsequent results in the evaluation phase, we made no attempts to correct and filter out misspelled words, except for the feature selection process described in Section 4.3. However, since OCR extracted documents preserve the text layout found in the original image, it was necessary to rejoin word fragments that had been hyphenated due to line breaking.

4.2. Grammar induction

In the case of completely labeled documents, it is possible to run the structure learning algorithm presented in Section 3.2. In figure 5 we show an example of induced HMM topology for the journal *The American Missionary*. This structure was extracted using 10 issues (year 1884) as a training set. Each vertex in the transition graph is associated with one HMM state and is labeled with the corresponding category index (see Table 1). Edges are labeled with the transition probability from source to target state, estimated in this case by counting state transitions during the state merging procedure (see Eq. (7)). These values are also used as initial estimates of $P(x^i|x^j)$ and subsequently refined by the EM algorithm. The associated stochastic grammar implies that valid sequences must start with the index page (class 1), followed by a page of general communications (class 8). Next state is associated with a page of an editorial article (2). Self transition here has a value of 0.91, meaning that with high probability the next page will belong to the editorial too. With lower probability (0.07) next page is one of "The South" survey (3) or (probability 0.008) "The Indians" (4) or "Bureau of Women's work" (6).

In figure 6 we show one example of induced HMM topology for journal *Scribners Monthly*, obtained from 12 training issues (year 1871). Although issues of *Scribners Monthly* are longer and the number of categories is comparable to those in the *American Missionary*, the extracted transition diagram in figure 6 is simpler than the one in figure 5. This reflects less variability in the sequential organization of articles in *Scribners Monthly*. Note that category 7 (Home and Society) is rare and never occurs in 1871.

4.3. Feature selection

Text pages were first preprocessed with common filtering algorithms including stemming and stop words removal. Still, the bag-of-words representation of pages leads to a very high-dimensional feature space that can be responsible of overfitting in conjunction to algorithms based on generative probabilistic models. Feature selection is a technique for limiting overfitting by removing non-informative words from documents. In our experiments, we performed feature selection using information gain (Yang and Pedersen, 1997). This criterion is often employed in different machine learning contexts. It measures the average number of bits of information about the category that are gained by including a word in a document. For each dictionary term w, the gain is defined as

$$G(w) = -\sum_{k=1}^{K} P(c^k) \log_2 P(c^k) + P(w) \sum_{k=1}^{K} P(c^k \mid w) \log_2 P(c^k \mid w) + P(\bar{w}) \sum_{k=1}^{K} P(c^k \mid \bar{w}) \log_2 P(c^k \mid \bar{w})$$

where $\bar{w}$ denotes the absence of word w. Feature selection is performed by retaining only the words having the highest average mutual information with the class variable. OCR errors, however, can produce very noisy features which may be responsible of poor performance

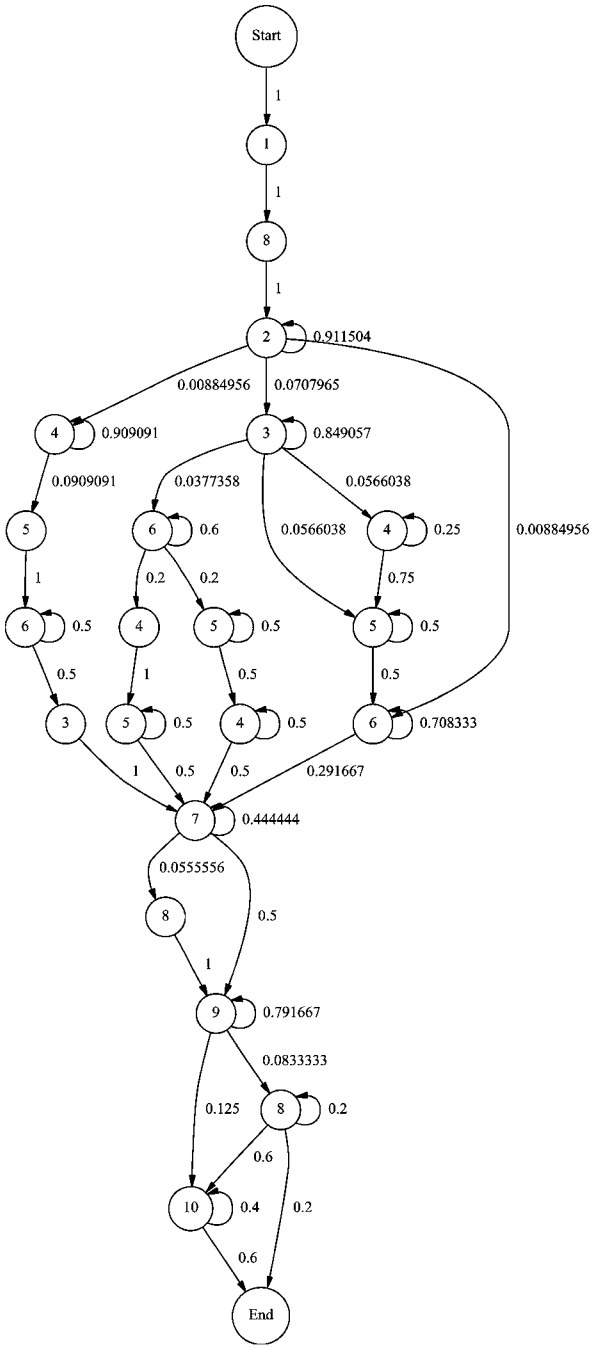

Figure 5. Data induced HMM topology for American Missionary, year 1884. Numbers in each node correspond to a page category (see Table 1).

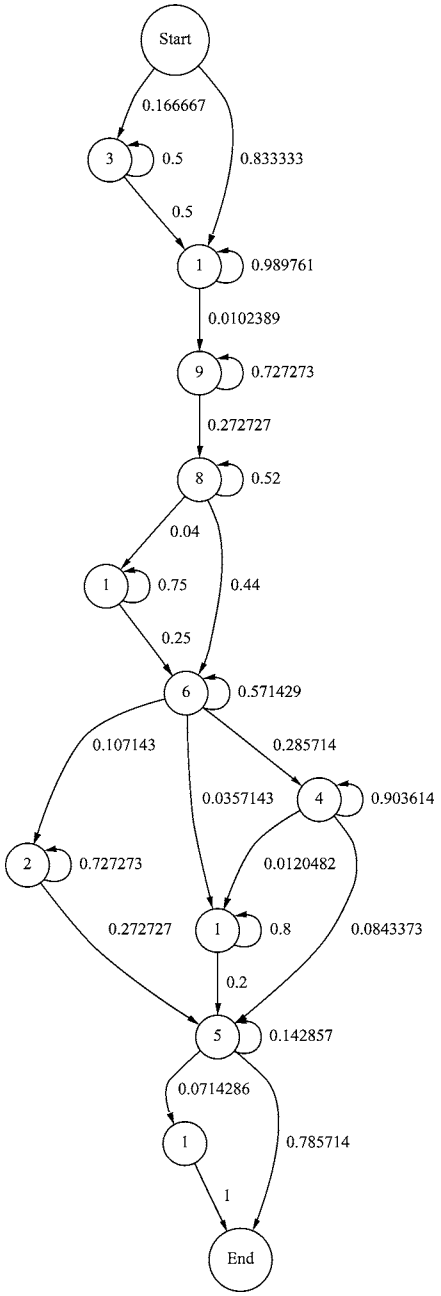

Figure 6. Data induced HMM topology for Scribners Monthly, year 1871. Numbers in each node correspond to a page category (see Table 2).

even if feature selection is performed. For this reason, it may be convenient to prune from the dictionary (before applying the information gain criterion) all the words occurring less than a given threshold h in the training set. Preliminary experiments showed that best performances are achieved by pruning words having less than $h = 10$ occurrences.

4.4. Accuracy comparisons

In the following we compare isolated page classification (using standard Naive Bayes) to sequential classification (using the proposed HMM architecture). Although classification accuracy could be estimated by fixing a split of the available data into a training and a test set, here we suggest a method that attempts to incorporate some peculiarities of digital libraries domain. In particular, hand-labeling of documents for the purpose of training is a very expensive activity and working with large training sets is likely to be unrealistic in practical applications. For this reason, in most experiments we deliberately used small fractions of the available data for training.

Moreover, there is a problem of temporal stability as the journal organization may change over time. In our test we attempted to address this aspect by assuming that training data is available for a given year and we decided to test generalization over journal issues published in different years. Splitting according to publication year can be an advantage for the training algorithm since it increases the likelihood that different issue organizations are represented in the training set.

The resulting method is related to k-fold cross-validation, a common approach for accuracy estimation that partitions the dataset into k subsets and iteratively use one subset for testing and the other $k - 1$ for training. In our experiments we reversed the proportions of data in the training and test sets, using all the journal issues in one year for training, and the remaining issues for testing. We believe that this setting is more realistic in the case of digital libraries.

In the following experiments, the HMM classifiers were trained by first extracting the transition structure, then initializing the parameters using Eqs. (6) and (7), and finally tuning the parameters using the EM algorithm. We found that the initial parameter estimates are very close to the final solution found by the EM algorithm. Typically, 2 or 3 iterations are sufficient for EM to converge.

4.4.1. American Missionary dataset. The results of the ten resulting experiments are shown in figure 7. The hybrid HMM classifier (performing sequential classification) consistently outperforms the plain Naive Bayes classifier working on isolated pages. The graph on the top summarizes results obtained without feature selection. Averaging the results over all the ten experiments, NB achieves 61.9% accuracy, while the HMM achieves 80.4%. This corresponds to a 48.4% error rate reduction. The graph on the bottom refers to results obtained by selecting the best 300 words according to the information gain criterion. The average accuracy in this case is 69.8% for NB and 80.6% for the HMM (a 35.7% error rate reduction). In both cases, words occurring less than 10 times in their training sets were pruned. When using feature selection, NB improves while the HMM performance is essentially the same. Moreover, the standard deviation of the accuracy is smaller for NB (2.8%, compared to 4.2% for the HMM). The larger variability in the case of the HMM is due to

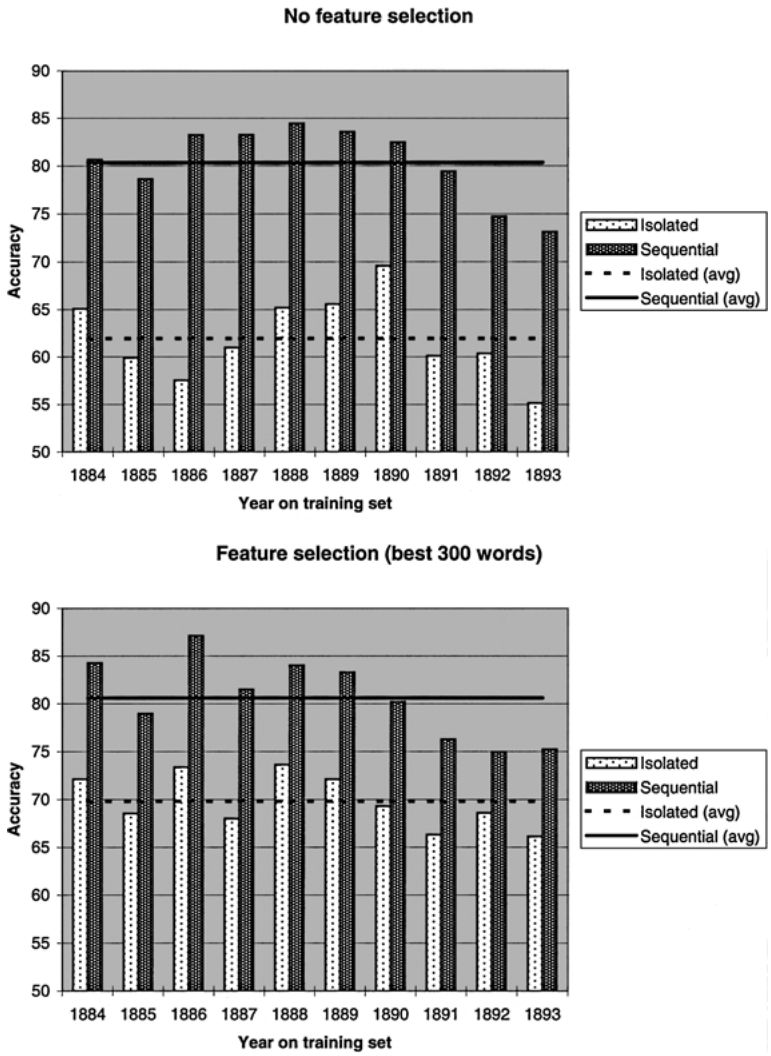

Figure 7. Isolated vs. sequential page classification on the American Missionary dataset. For each column, classifiers are trained on documents of the corresponding year and tested on all remaining issues.

the structure induction algorithm. In facts, the sequential organization of journal issues is temporally less stable than article contents.

4.4.2. Scribners Monthly dataset. Similar experiments have been carried out on the *Scribners Monthly* journal. Results using no feature selection are shown on the top of figure 8. The average accuracy is 81.0%, for isolated page classification and 89.6% for sequential classification (the error reduction is 42.5%). After feature selection, the average accuracy drops to 75.3% for the isolated page classifier, while it remains similar for the sequential classifier.

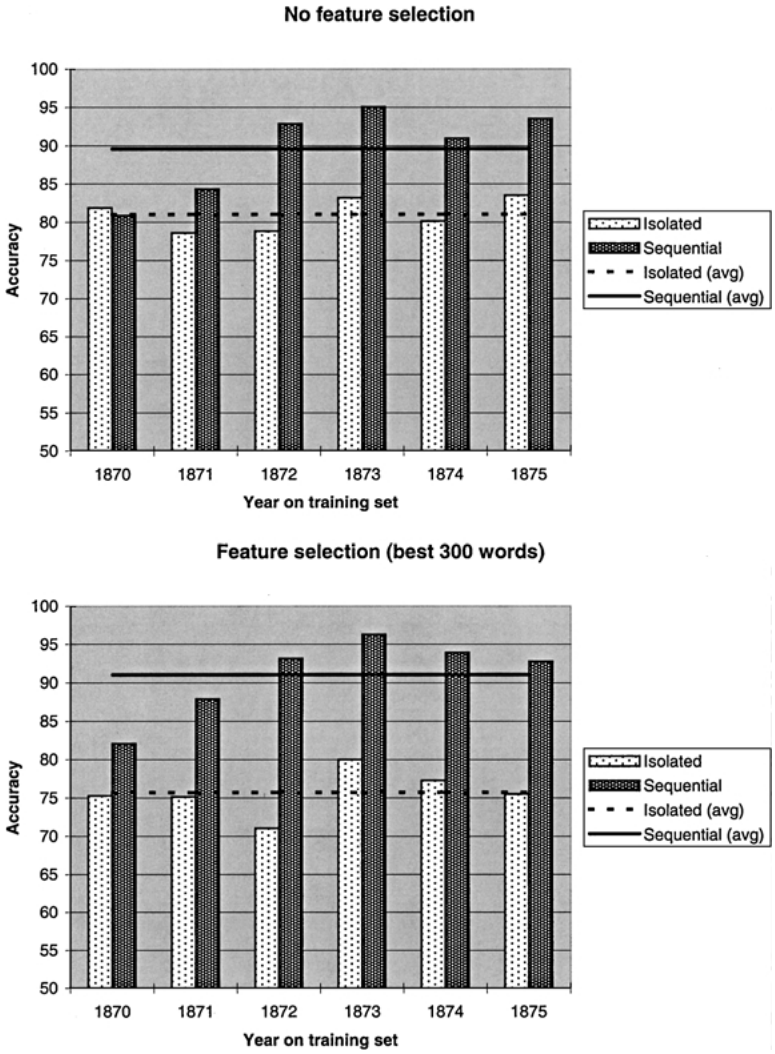

Figure 8. Isolated vs. sequential page classification on the Scribners Monthly dataset.

Noticeably, feature selection has different effects on the two datasets when coupled with the Naive Bayes classifier: it tends to improve accuracy for the *American Missionary* and tends to worsen for the *Scribners Monthly*. On the other hand, the HMM is almost insensitive to feature selection, in both datasets. This is apparently counterintuitive since the emission model is almost the same for the two classifiers (except for the EM tuning of emission parameters in the case of the HMM). However, it should be remarked that the Naive Bayes' final prediction is biased by the class prior (Eq. (16)) while the HMM's prediction is biased by extracted grammar (Eqs. (11)–(15)). The latter provides more robust information that effectively compensates for the crude approximation in the emission model, prescribing

conditional word independence. This robustness also affects positively performance if a suboptimal set of features is selected for representing document pages.

4.5. *Learning using ergodic HMMs*

The following experiments provide a basis for evaluating the effects of the structure learning algorithm presented in Section 3.2. In the present setting, we trained an ergodic HMM with ten states (each state mapped to exactly one class). Emission parameters were initialized using Eq. (6) while transition probabilities were initialized with random values. In this case the EM algorithm takes the full responsibility for extracting sequential structure from data. After training, arcs with associated probability less than 0.001 were pruned away.

The evaluation was performed using the *American Missionary* dataset, training on single years as in the previous set of experiments. As expected (see figure 9), results are worse than those obtained in conjunction with the grammar extraction algorithm. However, the trained HMM outperforms the Naive Bayes classifier also in this case.

4.6. *Effects of the training set size*

To investigate the effects of the size of the training set we propose a set of experiments alternative to those reported in Section 4.4. In these experiments we selected a variable number of sequences (journal issues) n for training (randomly chosen in the dataset) and tested generalization on all the remaining sequences. The accuracy is then reported as a function of n, after averaging over 20 trials (each trial with the same proportion of training and test sequences). All these experiments were performed on the *American Missionary* dataset. As shown in figure 10, generalization for both the isolated and the sequential classifier tends to saturate after about 15 sequences in the training set. This is slightly more

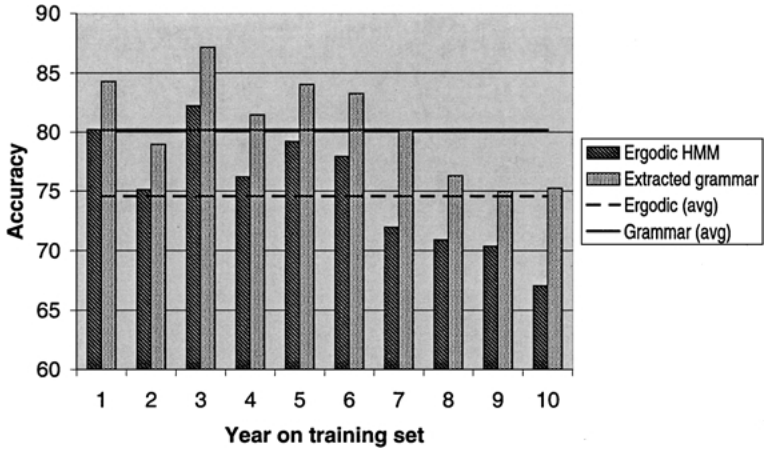

Figure 9. Comparison between the ergodic HMM and the HMM based on the extracted grammar.

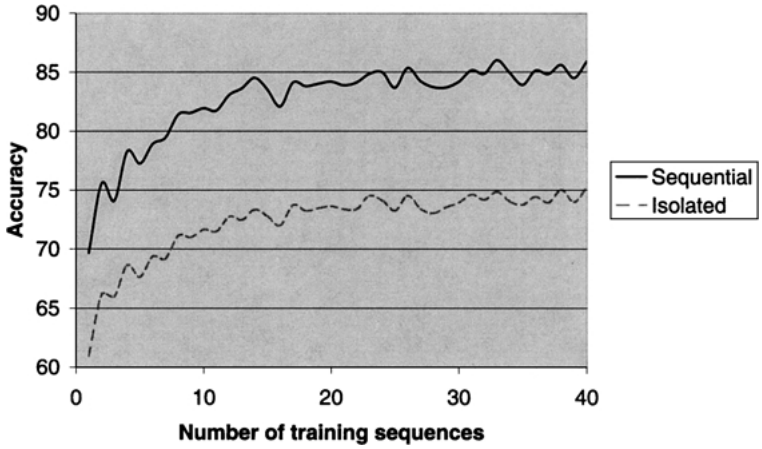

Figure 10. Learning curve for the sequential and the isolated classifiers.

than the average number of issues in a single year. The sequential classifier consistently outperforms the isolated page classifier.

4.7. Learning with partially labeled documents

Since labeling is an expensive human activity, we evaluated our system also when only a fraction of the training documents pages are labeled. In particular, we are interested in measuring the loss of accuracy due to missing page labels. Since structure learning is not feasible with partially labeled documents, we used in this case an ergodic (fully connected) HMM with ten states (one per class).

We have performed six different experiments on the *American Missionary* dataset, using different percentages of labeled pages. In all the experiments, all issues of year 1884 form the training set and the remaining issues form the test set. Table 3 shows detailed results of the experiment. Classification accuracy is reported for single classes and for the entire test set. Using 30% of labeled pages the HMM fails to learn a reliable transition structure and the Naive Bayes classifier (trained with EM as in Nigam et al. (2000)) obtains higher accuracy (Table 4). However, with higher percentages of known page labels the comparison favors again the sequential classifier. Using only 50% of labeled pages, the HMM outperforms the isolated page classifier that was trained on completely labeled data. With greater percentages of labeled documents, performances begin to saturate reaching a maximum of 80.24% when all the labels are known (this corresponds to the result obtained in Section 4.5).

5. Conclusions

We have presented a text categorization system for multi-page documents which is capable of effectively taking into account contextual information to improve accuracy with respect to traditional isolated page classifiers. Our method can smoothly deal with unlabeled pages within a document, although we have found that learning the HMM structure further

Table 3. Results achieved by the model trained by Expectation-Maximization, varying percentage of labeled documents.

	Percentage of labeled documents				
Category	30	50	70	90	100
Contents	100	100	100	100	100
Editorial	21.12	59.67	58.6	67.62	71.41
South	83.58	69.73	84.94	84.34	84.19
Indians	0	55.03	51.68	50.34	58.39
Chinese	27.45	83.66	76.47	75.82	75.16
Bur.WW	43.22	63.74	63	64.84	65.93
Child.P.	78.26	73.91	58.7	78.27	76.09
Communications	91.4	91.4	93.55	93.55	93.55
Receipts	89.27	98.68	97.36	98.31	98.31
Advertisements	69.77	93.02	90.7	90.7	90.7
Total	55.66	73.54	75.66	78.7	80.24

Table 4. Results achieved by Naive Bayes classifier trained by Expectation-Maximization, varying percentage of labeled documents.

	Percentage of labeled documents				
Category	30	50	70	90	100
Contents	100	100	98.82	100	100
Editorial	32.15	45.08	48.04	60.26	63.11
South	61.75	73.95	79.22	70.48	71.84
Indians	33.33	52.67	45.33	43.33	44.30
Chinese	73.03	75.66	66.45	68.42	60.78
Bur.WW	69.85	67.65	70.96	69.85	66.30
Child.P.	73.91	63.04	47.83	47.83	45.65
Communications	91.4	92.47	91.40	92.47	92.47
Receipts	97.74	98.11	98.11	98.30	98.31
Advertisements	58.14	62.79	55.81	55.81	62.79
Total	61.81	69.35	70.47	72.03	72.57

improves performance compared to starting from an ergodic structure. The system uses OCR extracted words as features. Clearly, richer page descriptions could be integrated in order to further improve performance. For example, most optical recognizers output information about the font, size, and position of text, that may be important to help discriminating between classes. Moreover, OCR text is noisy and another direction for improvement is to include more sophisticated feature selection methods, like morphological analysis or the use of n-grams (Cavnar and Trenkle, 1994; Junker and Hoch, 1998).

Another aspect is the granularity of document structure being exploited. Working at the level of pages is straightforward since page boundaries are readily available. However, actual category boundaries may not coincide with page boundaries. Some pages may contain portions of text belonging to different articles (in this case, the page would belong to multiple categories). Although this is not very critical for single-column journals such as the *American Missionary*, the case of documents typeset in two or three columns certainly deserves attention. A further direction of investigation is therefore related to the development of algorithms capable of performing automatic segmentation of a continuous stream of text, without necessarily relying on page boundaries.

Finally, text categorization methods that take document structure into account may be extremely useful for other types of documents natively available in electronic form, including web pages and documents produced with other typesetting systems. In particular, hypertexts (like most documents in the Internet) are organized as directed graphs, a structure that can be seen as a generalization of sequences. However, devising a classifier that can capture context in hypertexts by extending the architecture described in this paper is still an open problem: although the extension of HMMs from sequences to trees is straightforward (see e.g. Diligenti et al. (2001)), the general case of directed graphs is difficult because of the presence of cycles. Preliminary research in this direction (based on simplified models incorporating graphical transition structure) is presented in Diligenti et al. (2000) and Passerini et al. (2001).

Acknowledgments

We thank the Cornell University Library for providing us data collected within the Making of America project. This research was partially supported by EC grant # IST-1999-20021 under METAe project.

Notes

1. A related formulation would consist of assigning a global category to a whole multi-page document, but this formulation is not considered in this paper.
2. After observing the text.
3. A Bayesian network is an annotated graph in which nodes represent random variables and *missing* edges encode conditional independence statements amongst these variables. Given a particular state of knowledge, the semantics of belief networks determine whether collecting evidence about a set of variables does modify one's belief about some other set of variables (Jensen, 1996; Pearl, 1988).
4. We adopt the standard convention of denoting variables by uppercase letters and realizations by the corresponding lowercase letters. Moreover, we use the table notation for probabilities as in Jensen (1996); for example $P(X)$ is a shorthand for the table $[P(X = x^1), \ldots, P(X = x^r)]$ and $P(X, y \mid Z)$ denotes the two-dimensional table with entries $P(X = x^i, Y = y \mid Z = z^k)$.
5. Of course this does not mean that the *category* is independent of the context.

References

1. Bengio, Y. and Frasconi, P. (1995). An Input Output HMM Architecture. In G. Tesauro, D. Touretzky, and T. Leen (Eds.), *Advances in Neural Information Processing Systems vol. 7* (pp. 427–434). Cambridge, MA: MIT Press.

2. Bicknese, D.A. (1998). Measuring the Accuracy of the OCR in the Making of America. Report available at `moa.umdl.umich.edu/moaocr.html`.
3. Cavnar, W. and Trenkle, J. (1994). *N*-Gram Based Text Categorization. In *Proc. of the 3rd Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas (pp. 161–175).
4. Charniak, E. (1993). *Statistical Language Learning*. Cambridge, MA: MIT Press.
5. Cohen, W.W. (1995). Text Categorization and Relational Learning. In A. Prieditis and S.J. Russell (Eds.), *Proc. of the 12th International Conference on Machine Learning*, Lake Tahoe, California (pp. 124–132).
6. Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum-Likelihood from Incomplete Data via the EM Algorithm. *Journal of Royal Statistical Society B*, 39, 1–38.
7. Diligenti, M., Coetzee, F., Lawrence, S., Giles, L., and Gori, M. (2000). Focus Crawling by Context Graphs. In *Proc. of the 26th International Conference on Very Large Databases* (pp. 527–534).
8. Diligenti, M., Frasconi, P., and Gori, M. (2001). Image Document Categorization Using Hidden Tree-Markov Models and Structured Representations. In S. Singh, N.A. Murshed, and W. Kropatsch (Eds.), *Proc. 2nd International Conference on Advances in Pattern Recognition*, volume 2013 of *LNCS* (pp. 147–156). Berlin: Springer.
9. Freitag, D. and McCallum, A. (2000). Information Extraction with HMM Structures Learned by Stochastic Optimization. In *Proc. of the 12th AAAI Conference*, Austin, TX (pp. 584–589).
10. Heckerman, D. (1997). Bayesian Networks dor Data Mining. *Data Mining and Knowledge Discovery*, 1(1), 79–120.
11. Jensen, F.V. (1996). *An Introduction to Bayesian Networks*. New York: Springer Verlag.
12. Joachims, T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In C. Ndellec and C. Rouveirol (Eds.), *Proceedings of the European Conference on Machine Learning* (pp. 137–142), Berlin: Springer.
13. Joachims, T. (1999). Transductive Inference for Text Classification using Support Vector Machines. In I. Bratko and S. Dzeroski (Eds.), *Proc. of the 16th International Conference on Machine Learning* (pp. 200–209).
14. Junker, M. and Hoch, R. (1998). An Experimental Evaluation of OCR Text Representations for Learning Document Classifiers. *International Journal on Document Analysis and Recognition*, 1(2), 116–122.
15. Kalt, T. (1996). A New Probabilistic Model of Text Classification and Retrieval. CIIR TR98-18, University of Massachusetts. url: `ciir.cs.umass.edu/publications/`.
16. Koller, D. and Sahami, M. (1997). Hierarchically Classifying Documents using Very Few Words. In D.H. Fisher (Ed.), *Proc. of the 14th International Conference on Machine Learning* (pp. 170–178).
17. Lewis, D. and Gale, W. (1994). A Sequential Algorithm for Training Text Classifiers. In W.B. Croft and C.J. van Rijsbergen (Eds.), *Proc. of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval* (pp. 3–12).
18. Lewis, D.D. and Ringuette, M. (1994). Comparison of Two Learning Algorithms for Text Categorization. In *Proc. of the 3rd Annual Symposium on Document Analysis and Information Retrieval* (pp. 81–93).
19. Lucke, H. (1995). Bayesian Belief Networks as a Tool for Stochastic Parsing. *Speech Communication*, 16, 89–118.
20. McCallum, A., Nigam, K., Rennie, J., and Seymore, K. (2000). Automating the Construction of Internet Portals with Machine Learning. *Information Retrieval Journal*, 3, 127–163.
21. Mitchell, T. (1997). *Machine Learning*. New York: McGraw-Hill.
22. Ng, H.T., Goh, W.B., and Low, K.L. (1997). Feature Selection, Perceptron Learning, and a Usability Case Study for Text Categorization. In *Proc. of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 67–73).
23. Nigam, K., McCallum, A., Thrun, S., and Mitchell, T. (2000). Text Classification from Labeled and Unlabeled Documents using EM. *Machine Learning*, 39(2/3), 103–134.
24. Passerini, A., Frasconi, P., and Soda, G. (2001). Evaluation Methods for Focused Crawling. In *Proc. of the 7th Conference of the Italian Association for Artificial Intelligence*, Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag.
25. Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann.

26. Rabiner, L.R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2), 257–286.
27. Shaw, E.J. and Blumson, S. (1997). Online Searching and Page Presentation at the University of Michigan. *D-Lib Magazine*, July/August 1997. URL: `www.dlib.org/dlib/july97/america/ 07shaw.html`.
28. Smyth, P., Heckerman, D., and Jordan, M.I. (1997). Probabilistic Independence Networks for Hidden Markov Probability Models. *Neural Computation*, 9(2), 227–269.
29. Stolcke, A. and Omohundro, S. (1993). Hidden Markov Model Induction by Bayesian Model Merging. In S.J. Hanson, J.D. Cowan, and C.L. Giles (Eds.), *Advances in Neural Information Processing Systems*, *vol. 5* (pp. 11–18). San Mateo, CA: Morgan Kaufmann.
30. Yang, Y. and Chute, C.G. (1994). An Example-Based Mapping Method for Text Classification and Retrieval. *ACM Transactions on Information Systems*, 12(3), 252–277.
31. Yang, Y. and Pedersen, J.P. (1997). A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the 14th International Conference on Machine Learning* (pp. 412–420).