2009 10th International Conference on Document Analysis and Recognition

Using Kernel Density Classifier with Topic Model and Cost Sensitive Learning for Automatic Text Categorization

Dwi Sianto Mansjur, Ted S. Wada, Biing Hwang Juang
Center for Signal and Image Processing
Georgia Institute of Technology
Atlanta, GA 30318, USA

Abstract

This paper proposes a novel framework for automatic text categorization problem based on the kernel density classifier. The overall goal is to tackle two main issues in automatic text categorization problems: the interpretability and the performance. Specifically, to solve the interpretability issue, the Latent Semantic Analysis technique is used to construct a topic space, in which each dimension represents a single topic. The text features are extracted directly from this topic space. To solve the performance issue, classifiers' parameters are optimized for either cost-sensitive or non-cost-sensitive categorization. We have experimentally evaluated the proposed framework by using a corpus of twenty newsgroups. The experimental results confirm the effectiveness of the framework to utilize the features from the topic model for cost-sensitive categorization.

1. Introduction

Most text categorization systems nowadays are using the Naive Bayes (NB) classifiers based on either a Bernoulli or multinomial document event model [5]. Nevertheless, there are still many open issues associated with using the NB classifiers, such as their performances not being as good as the Support Vector Machine (SVM) classifiers [5], or their interpretability being limited due to the use of single word as a feature component. To solve the interpretability issue, some researchers have proposed using either language-model-based or the tree-augmented NB classifiers [2]. The introduction of SVM has offered a refreshing view on text categorization problems because direct minimization of a classification loss became possible. However, SVM solutions are difficult to interpret due to the absence of the class distribution functions and its inability to quantify the cost of making a classification error. We believe that a good text categorization system should include combinations of multiple words as features, and that a good classifier should take into account both the classification loss and the class distribution functions.

In order to tackle the interpretability issue in the text categorization problem, a topic modeling approach called Latent Semantic Analysis (LSA) [1] is often used. Other topic modeling techniques, such as Probabilistic Latent Semantic Analysis (PLSA) [3] or Latent Dirichlet Allocation (LDA) may substitute for LSA. Topic modeling approaches are different from either the language-model based or the tree-augmented NB classifiers because it is intended to capture hidden dependencies among multiple words [1].

In this paper, we propose a framework for the automatic text categorization problem using the Kernel Density Classifier (KDC) along with a suitable topic model derived from LSA and a *cost-sensitive* paradigm [6]. To the best of our knowledge, this is the first attempt in using such a framework. The functional form of the KDC is not only suitable for estimating the true class distribution but also for incorporating unlabeled dataset (semi-supervised learning) [9]. The KDC is different from the conventional SVM classifier because it relies on the estimation of class distribution functions. However, the estimated distribution functions may be inaccurate due to the limited training data size. Therefore, we utilize a cost-sensitive optimization procedure to enhance the quality of density estimation and its effect on the classification accuracies. The optimization procedure is suitable for cost-sensitive learning, which includes *non-cost-sensitive* learning as a special case [6].

This paper is organized as follows. In Section 2, we review the topic model extraction procedure based on LSA. In Section 3, we introduce the Bayes risk objective function and the proposed KDC approach. In Section 4, we review the procedure to implement the Bayes risk objective function suitable for a cost minimization procedure. In Section 5, we demonstrate how to obtain the cost minimization procedure. In Section 6, we present the experimental results from using a standard text categorization corpus. Conclusions and future work are provided in Section 7.

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.145

2 Topic Model

In this paper, we use LSA to obtain a topic model. The main idea of LSA is its bag-of-words document representation that preserves enough information relevant to semantic categorization [1]. Suppose that we have a collection $D = \{d_1, \cdots, d_N\}$ of N documents and a lexicon $W = \{t_1, \cdots, t_T\}$ with T words. Neglecting the order of words in each document, one may represent the collection of documents as a $T \times N$ co-occurrence matrix $A = \{a_{ij}\}$, where a_{ij} denotes the number of occurrence of the word t_i in the document d_j. That is, each document is represented in the word-space by the j^{th} column vector of the matrix A.

In a practical scenario, the co-occurrence matrix is usually sparse, and the documents may have different lengths, i.e. different number of words. To account for these difficulties, we propose using the Term-Frequency Inverse-Document Frequency (tfidf) weighting

$$tfidf_{ij} = \left(\frac{a_{ij}}{\sum_k a_{kj}}\right) \times \log\left(\frac{N}{|\{d_j : t_i \in d_j\}|}\right), \quad (1)$$

where $tfidf_{ij}$ is the i^{th} row and j^{th} column entry of the new co-occurrence matrix $\hat{A} = \{tfidf_{ij}\}$, and $|\{d_j : t_i \in d_j\}|$ is the number of documents containing the word t_i.

By using the Singular Value Decomposition (SVD) technique, the co-occurrence matrix $\hat{A}$ is decomposed as $\hat{A} = U\Sigma V'$, where Σ is the diagonal matrix containing the singular values of $\hat{A}$, U and V are the orthonormal matrices consisting of eigenvectors from the word-space and the document-space, respectively, and V' indicates the transpose of V. By retaining the K largest singular values in Σ_K and the corresponding eigenvectors in $\tilde{U}$ and $\tilde{V}$, $\hat{A}$ is approximated as $\tilde{A} = \tilde{U}\Sigma_K\tilde{V}'$ that represents the K-most latent ensembles of words and documents.

The overall procedure is summarized as follows: given a document d_n in the word-space of A, the tfidf transformation is applied to obtain $\hat{d}_n$ of $\hat{A}$, and the corresponding representation of the document in the latent semantic space $\tilde{A}$ is obtained from the transformation $\tilde{d}_n = \Sigma_K^{-1}\tilde{U}'\hat{d}_n$.

3 Kernel Density Classifier

The main task in a text categorization system is to classify an unlabeled document, denoted by $\tilde{d}_n$, into one of the M classes. The categorization system is represented by a decision function $\mathbb{C}$ that maps $\tilde{d}_n$ into a class identity C_i, where $i \in I_M = \{1, \cdots, M\}$. Each decision is associated with a cost ϵ_{ij} in an $M \times M$ cost matrix, where $i, j \in I_M$. The entry ϵ_{ij} indicates the cost of classifying a text document from the j^{th} class into the i^{th} class. The overall system performance is defined in terms of the *expected loss*

$$\mathcal{L} = \int R(\mathbb{C}(D)|D)p(D)dD. \quad (2)$$

In the text categorization context, the Bayes decision rule can be stated as

$$\tilde{d}_{N+1} \in C_i \text{ if } i = \arg\min_k R(C_k|\tilde{d}_{N+1}) \quad (3)$$

for a given document $\tilde{d}_{N+1}$. The conditional cost of making a decision $\mathbb{C}(\tilde{d}_{N+1}) = C_i$ can be defined as $R(C_i|\tilde{d}_{N+1}) = \sum_{j=1}^{M} \epsilon_{ij} P(C_j|\tilde{d}_{N+1})$. The posterior probability $P(C_j|\tilde{d}_{N+1})$ can be computed by using the Bayes' formula, i.e. $P(C_j|\tilde{d}_{N+1}) = \frac{P(\tilde{d}_{N+1}|C_j)}{p(\tilde{d}_{N+1})}P(C_j)$, where $P(\tilde{d}_{N+1}|C_j)$, $p(\tilde{d}_{N+1})$, and $P(C_j)$ denote the likelihood of class C_j, the probability of document $\tilde{d}_{N+1}$, and the prior for class C_j, respectively. The likelihood $P(\tilde{d}_{N+1}|C_j)$ can further be parameterized as

$$P(\tilde{d}_{N+1}|C_j; \Lambda_{C_j}) = \frac{1}{N_j}\sum_{n=1}^{N_j}\prod_{k=1}^{K}\frac{1}{h_n^{(k)}}\mathcal{N}\left(\frac{\tilde{d}_{N+1} - \tilde{d}_n^{(k)}}{h_n^{(k)}}\right), \quad (4)$$

where N_j is the number of training documents from class C_j, $\mathcal{N}$ is the Gaussian kernel function, and $h_n^{(k)}$ is the bandwidth. For simplicity, we assign a kernel to each document $\tilde{d}_n^{(k)}$ such that $\Lambda_{C_j} = \{h_n^{(k)}\}$ and assume mutual independence between K dimensions of the latent word-space. Moreover, $h_n^{(k)}$ may be initialized from the training data by using several methods, e.g. Silverman Rule of Thumb (ROT), Maximum Likelihood Cross-Validation (MLCV), or Nearest Neighbor Estimation (NNE) [7, 8]. To obtain the initial bandwidth parameters, we use the NNE approach to obtain the nearest neighbor distance among samples from all classes. Such an approach gives large bandwidth for a sparse region and small bandwidth for a dense region.

4 Cost Minimization Procedure

In practice, the Bayes risk in Eq.(2) can only be approximated from the available training data. Juang et.al [4] had proposed the Minimum Classification Error (MCE) framework with the objective of minimizing the zero-one classification error-cost. Recently, Fu et.al [6] has provided a novel MCE formulation for learning with non-uniform error-cost, which will be used in this paper to minimize the Bayes risk.

Let $g_i(X; \Lambda_{C_i}) \geq 0$ be the discriminant function for class C_i, where $i \in I_M$ and Λ_{C_i} is the parameter set that defines the class C_i. The classification decision is reached according to the following rule:

$$\mathbb{C}(\tilde{d}_n) = i \text{ if } i = \arg\max_k g_k(\tilde{d}_n; \Lambda_{C_k}), \quad (5)$$

where i is the class label of the largest discriminant function evaluated on document $\tilde{d}_n$. Because the classification

decision rule in Eq.(5) is based on the largest value of the discriminant function whereas the Bayes' rule in Eq.(3) is based on the smallest value of the risk function, we then modify the discriminant function as

$$g_k(\tilde{d}_n;\Lambda_{C_k}) = \exp\{-R(C_k|\tilde{d}_n)\} \quad (6)$$
$$= \exp\{-\sum_{j\in I_M}\frac{\epsilon_{kj}P(\tilde{d}_n|C_j)P(C_j)}{\sum_{s\in I_M}P(\tilde{d}_n|C_s)P(C_s)}\},$$

where $P(\tilde{d}_n|C_j)$ is the likelihood function for class C_j and $P(C_j)$ is the class prior.

Given a set of labeled training documents, the expected risk function in Eq.(2) may be estimated empirically as

$$L = \frac{1}{N}\sum_{n=1}^{N}\sum_{i\in I_M}\sum_{j\in I_M}\epsilon_{ij} \mathbf{1}\{j=j_n\}\mathbf{1}\{i=\arg\max_k g_k(\tilde{d}_n;\Lambda_{C_k})\}. \quad (7)$$

The definition Eq.(7) uses two indicator functions, where an indicator function assumes the value of 1 if the argument is true and 0 otherwise. That is, the first indicator function in Eq.(7) specifies that observation $\tilde{d}_n$ belongs to class C_j, i.e. $\mathbf{1}\{j=j_n\} = \mathbf{1}\{\tilde{d}_n \in C_j\}$, and the second indicator function in Eq.(7) denotes the decision rule in Eq.(5).

Fu et.al [6] have proposed a novel procedure to convert the empirical risk function into a smooth function

$$\hat{L} \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{j\in I_M}\sum_{i\in I_M}\epsilon_{ij}\left(\frac{g_i(\tilde{d}_n;\Lambda_{C_i})}{G(X;\Lambda)}\right)^{\beta}\mathbf{1}\{\tilde{d}_n\in C_j\}, \quad (8)$$

where $G(\tilde{d}_n;\Lambda) = [\sum_{i\in I_M} g_i^{\eta}(\tilde{d}_n;\Lambda_{C_i})]^{1/\eta}$. Hereafter, we simplify the notation and refer to $g_i(\tilde{d}_n;\Lambda_{C_i})$ and $G(\tilde{d}_n;\Lambda)$ as g_i and G, respectively. Note that the design parameters η and β are usually set to some positive values depending on the intended application, and the approximation

$$\left(\frac{g_i}{G}\right)^{\beta} \approx \begin{cases} 1, & \text{if } g_i = \max_k g_k, \\ 0, & \text{otherwise,}\end{cases}$$

holds as $\eta\to\infty$ and $\beta\to\infty$.

The above formulation departs from the conventional error-rate minimization [4] because the errors are allowed to have different costs, e.g., misclassifying class 1 as class 3 may cost more than misclassifying class 1 as class 2. Moreover, the objective function in Eq.(8) is novel because it emphasizes the contribution of each training sample that comes from class C_j with respect to all possible decision $i\in I_M$ of the classifier. This indicates that the discriminant functions $\{g_i; i\in I_M\}$ for all classes are optimized for each of the training samples during the parameter optimization procedure.

5 Parameter Optimization Procedure

The optimization procedure is intended to correct inaccurate estimation of bandwidth values. The discriminant function g_i is set to be $\exp(\{-R(C_i|d_n)\})$ by using the definitions in Eq.(4) and Eq.(6). During the optimization procedure, the kernel weights and centroids are fixed, and only the bandwidth parameters are adjusted, i.e. $\Lambda_{C_i} = \{h_n^{(k)}; n\in\{1,\cdots,N_j\}, k\in\{1,\cdots,K\}\}$. All bandwidth values must be positive throughout the learning process, thus the following parameter transformation is conducted: $h_n^{(k)} \to \tilde{h}_n^{(k)} = \log(h_n^{(k)})$.

In order to optimize the objective function, we use a gradient-descent algorithm on each of the training data. Specifically, the cost associated with one training sample $\tilde{d}_n$ is defined by Eq.(8) as $\hat{L} \approx \frac{1}{N}\sum_{n=1}^{N}\ell_n^{(j)}$, where $\ell_n^{(j)} = \sum_{i\in I_M}\epsilon_{ij}\left(\frac{g_i}{G}\right)^{\beta}$. The corrective bandwidth learning algorithm is based on the following update rule:

$$\tilde{h}_n^{s+1} = \tilde{h}_n^{s} - \varepsilon^s U^s \bigtriangledown \ell(\tilde{h}_n^s), \quad (9)$$

where $\tilde{h}_n = \{\tilde{h}_n^{(k)}; k\in\{1,\cdots,K\}\}$, s is the iteration index, U^s is some positive definite matrix, ε^s is the step size, and $\bigtriangledown\hat{\ell}(\tilde{h}_n^s)$ is the gradient of the objective function. By denoting $P(\tilde{d}_n|C_i;\Lambda_{C_i})$ and $P(C_i)$ as b_i and P_i, respectively, the gradient is computed by using the following formulas:

$$\frac{\partial\ell_n^{(j)}}{\partial b_m} = \frac{\beta P_m}{\sum_{r=1}^{M}b_r P_r}\left\{\left(\frac{-\sum_{k=1}^{M}\epsilon_{kj}(\epsilon_{km}+\log(g_k))g_k^{\beta}}{G^{\beta}}\right) + \left(L_n\frac{\sum_{k=1}^{M}(\epsilon_{km}+\log(g_k))g_k^{\eta}}{G^{\eta}}\right)\right\}, \quad (10)$$

$$\frac{\partial b_j}{\partial\tilde{h}_n^{(k)}} = \frac{(2\pi)^{-K/2}}{N_j}\left(\prod_{k=1}^{K}h_n^{(k)}\right)^{-1}\left[\left(\frac{\tilde{d}^{(k)}-\tilde{d}_n^{(k)}}{h_n^{(k)}}\right)^2 - 1\right] exp\left\{-\frac{1}{2}\sum_{k=1}^{K}\left(\frac{\tilde{d}^{(k)}-\tilde{d}_n^{(k)}}{h_n^{(k)}}\right)^2\right\}\mathbf{1}(\tilde{d}_n^{(k)}\in C_j). \quad (11)$$

The convergence properties of the corrective bandwidth learning algorithm were studied in the literature under the name of stochastic approximation [4]. After the learning process takes place, the following inverse transformation is applied to obtain new bandwidth parameters: $\tilde{h}_n^{(k)} \to h_n^{(k)} = \exp(\tilde{h}_n^{(k)})$.

6 Experiments

To verify the effectiveness of the proposed KDC framework, text categorization experiments were conducted on a dataset from 20 newsgroups. The original

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

(a) Average of cost matrices used in non-cost-sensitive learning.

$$\begin{bmatrix} 0 & 70.9 & 9.9 & 74.0 & 91.4 & 31.1 & 27.7 \\ 70.9 & 0 & 8.9 & 45.2 & 45.4 & 17.7 & 22.4 \\ 9.9 & 8.9 & 0 & 32.0 & 7.2 & 34.4 & 5.2 \\ 74.0 & 45.2 & 32.0 & 0 & 37.3 & 49.8 & 18.5 \\ 91.4 & 45.4 & 7.2 & 37.3 & 0 & 20.7 & 46.7 \\ 31.1 & 17.7 & 34.4 & 49.8 & 20.7 & 0 & 14.1 \\ 27.7 & 22.4 & 5.2 & 18.5 & 46.7 & 14.1 & 0 \end{bmatrix}$$

(b) Average of cost matrices used in cost-sensitive learning.

Table 1. Average of cost matrices for (a) non-cost-sensitive and (b) cost-sensitive learning.

		comp	rec	sci	talk	alt	misc	soc	micro-avg	macro-avg
SVM	Precision	64.7(11.0)	**74.9(13.6)**	70.4(9.2)	**76.0(3.9)**	**72.5(11.3)**	66.2(6.8)	54.1(19.8)	68.9(3.0)	68.4(2.8)
	Recall	62.7(13.8)	**70.9(9.5)**	**86.0(9.5)**	**89.3(3.7)**	67.9(9.7)	**80.9(7.2)**	47.7(10.2)	68.9(3.0)	**72.2(2.1)**
Baseline KDC	Precision	58.0(12.9)	59.4(6.0)	73.5(9.5)	73.3(5.4)	60.9(9.0)	66.0(10.7)	42.8(6.0)	62.1(4.2)	62.0(4.1)
	Recall	58.0(9.9)	59.4(9.1)	73.5(11.2)	73.3(6.8)	60.9(7.4)	66.0(4.8)	42.8(11.2)	62.1(4.2)	61.5(4.0)
Optimized KDC	Precision	**67.6(11.1)**	64.7(11.9)	**76.0(8.5)**	72.2(7.7)	70.6(9.2)	**77.3(7.8)**	**64.4(9.8)**	**69.4(2.3)**	**70.4(2.3)**
	Recall	**64.0(8.3)**	66.9(8.2)	78.1(9.1)	81.1(5.0)	**70.2(7.4)**	73.7(10.7)	**48.1(15.6)**	**69.4(2.3)**	68.9(2.5)

Table 2. Average precision (standard deviation) and recall (standard deviation) from using non-cost-sensitive classification. Boldface value indicates the best performance.

dataset has approximately 20,000 news documents grouped into 20 hierarchical categories that were grouped into 7 distinctive categories, (i.e., "comp", "rec", "sci", "talk", "alt", "misc", and "soc"). For example, the category "comp" was a combination of the categories "comp.graphics", "comp.sys.mac.hardware", "comp.windows.x", "comp.os.ms-windows.misc", and "comp.sys.ibm.pc.hardware". There are 18,828 documents left after the removal of duplicated documents.

To compute the error-cost and the error-rate (accuracy), a 10-fold validation technique was used. Each experiment was carried out using 900 documents randomly selected from 18,828 news documents, which were then further divided equally into training, validation, and testing documents. To find the word feature, stopword removal and word stemming were applied to the corpus. For each experiment, the vocabulary had 1,000 words that were obtained from 300 training documents. The topic models were extracted from the word counts as specified in Section 2. All experiments were conducted using 30 topics ($K = 30$).

The "baseline KDC" refers to classifier in the form of Eq.(4), and the bandwidth parameters were obtained by using the NNE approach. The bandwidth parameters in the "optimized KDC" were obtained based on the objective function in Eq.(8), which allows for either *non-cost-sensitive* or *cost-sensitive* optimization. For comparison purpose, the conventional SVM classifier also was obtained by using the Libsvm package. To ensure fairness, the training, validation and testing documents were the same for both the KDC and the SVM classifier. For the SVM classifier, the data were normalized between -1 and $+1$, as is commonly done in practice. Linear kernel for the SVM classifier was used, and 15 possible cost parameter $\hat{C} = [2^{12}, 2^{11}, \cdots, 2^{-2}]$ were used in searching for the best kernel parameter $\hat{C}$.

For *non-cost-sensitive* categorization, the conventional zero-one cost matrix (shown in Table 1(a)) was used; which means that all errors have the same cost. As observed from the experimental results in Table 2, the baseline KDC has relatively lower performance when compared to the SVM classifier with linear kernel. On the other hand, the cost-optimized KDC exhibits performance comparable with the SVM classifier in terms of the *micro-average* and *macro-average* precision and recall. For such a zero-one cost matrix, the optimized KDC has the same objective function as the conventional MCE [4] that minimizes the expected loss or the error-rate. In addition, the interpretation of the micro- and macro-average precision and recall remain the same since it is assumed for their calculation that all classes are weighted equally and that all errors have the same cost.

However, depending on applications, the error-cost assignments are not always zero-one. For example, some doctors in medical diagnosis may use the cost of 100 for erroneously diagnosing a patient to be healthy and 1 for mistakenly diagnosing a healthy person as being sick, whereas other doctors may use the cost of 1000 for the first type of mistake and 10 for the second type of mistake. Such kind of cost assignment can be named "user-specified" cost assignment [6]. Another way of assigning a cost matrix is "data-driven", for which the cost matrix is proportional to the statistics of the data, e.g. the class prior.

The data-driven approach was used for testing *cost-*

$$\begin{bmatrix} 15.6 & 4.3 & 1.3 & 3.9 & 6.7 & 2.2 & 2.0 \\ 6.3 & 26.8 & 2.0 & 3.6 & 2.1 & 0.5 & 2.7 \\ 3.1 & 2.6 & 28.4 & 7.2 & 2.9 & 4.7 & 2.3 \\ 3.1 & 2.3 & 4.3 & 20.1 & 1.2 & 3.0 & 0.4 \\ 7.6 & 1.9 & 1.6 & 1.3 & 26.3 & 2.1 & 3.8 \\ 1.2 & 1.8 & 4.2 & 3.4 & 0.6 & 30.6 & 0.4 \\ 2.9 & 3.7 & 0.9 & 1.3 & 3.7 & 0.9 & 34.2 \end{bmatrix}$$

(a) Average confusion matrix of baseline KDC models.

$$\begin{bmatrix} 5.3 & 0.8 & 0 & 0 & 1.0 & 0.4 & 0.3 \\ 0.4 & 10.4 & 0.3 & 0.4 & 0.2 & 0.2 & 0.6 \\ 21.1 & 20.3 & 31.8 & 20.3 & 24.7 & 19.5 & 21.8 \\ 0.5 & 0.6 & 0.6 & 5.9 & 0 & 0.6 & 0.1 \\ 1.2 & 0.6 & 0.1 & 0.2 & 8.1 & 0.9 & 0.5 \\ 3.7 & 3.5 & 3.1 & 4.7 & 3.2 & 15.2 & 2.8 \\ 7.6 & 7.2 & 6.8 & 9.3 & 6.3 & 7.2 & 19.7 \end{bmatrix}$$

(b) Average confusion matrix of cost-optimized KDC models.

Table 3. Average of confusion matrices of (a) baseline and (b) cost-optimized models for cost-sensitive classification.

	Accuracy	Error-cost
Baseline KDC	62.1(4.2)	16.5(2.8)
Optimized KDC	32.1(0.1)	9.9 (1.3)

Table 4. Accuracy (standard deviation) and error-cost (standard deviation) of baseline and cost-optimized KDC.

sensitive categorization. Specifically, the cost matrix was derived by modeling each of the class distribution as a Gaussian distribution i.e. $C_i \sim \mathcal{N}_i(\mu_i, \Sigma_i)$. By using the Kullback-Leibler divergence between two class distributions, determined by $kl(\mathcal{N}_i, \mathcal{N}_j) = 0.5 \times \{\log(|\Sigma_j|/|\Sigma_i|) + (\mu_j - \mu_i)'\Sigma_j(\mu_j - \mu_i) + tr(\Sigma_j^{-1}\Sigma_i) - K\}$, where $|\Sigma_j|$ is the determinant of matrix Σ_j and $tr(\cdot)$ indicates the trace of a matrix, the cost ϵ_{ij} in Eq.(8) was defined as $\epsilon_{ij} \in [0, 100 \times \{kl(\mathcal{N}_i||\mathcal{N}_j) + kl(\mathcal{N}_j||\mathcal{N}_i)\}^{-1}]$. That is, errors are more likely to occur when the class distributions are closer to each other and thus should be penalized more than when the distributions are farther apart. The corresponding cost matrix is shown in Table 1(b). The cost-sensitive categorization results are summarized in terms of the confusion matrix in Table 3 and in terms of the total confusion cost in Table 4. We can observe from Table 4 that both the accuracy and the error-cost have decreased significantly for the cost-optimized KDC compared to the baseline KDC. The interpretation is that in the cost-sensitive paradigm, it is acceptable for accuracy to decrease as long as the cost of error is optimized according to the user-specified cost matrix.

7 Conclusion

In this paper, we introduce a novel framework for the automatic text categorization problem based on the Kernel Density Classifier (KDC). The implementation of KDC with the features from a topic model derived from the Latent Semantic Analysis (LSA) procedure ensures the interpretability of the resulting models. Furthermore, the framework offers a way to minimize the expected total cost, thus it is suitable for either *non-cost-sensitive* or *cost-sensitive* categorization. For the cost-insensitive case, our experimental results show comparable performances of the optimized KDC and the Support Vector Machine (SVM) classifier. For the cost-sensitive case, the optimized KDC may or may not result in lower accuracy compared to the non-optimized KDC since the primary objective is the optimization of the final cost. In future work, we would like to explore using other types of cost matrix and also using unlabeled data in the proposed framework to improve the overall categorization procedure.

References

[1] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. L, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

[2] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163, 1997.

[3] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM Press New York, NY, USA, 1999.

[4] B.-H. Juang, W. Hou, and C.-H. Lee. Minimum classification error rate methods for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 5(3):257–265, May 1997.

[5] A. Kibriya, E. Frank, B. Pfahringer, and G. Holmes. Multinomial naive bayes for text categorization revisited. In *Proceedings of the 17th Australian Joint Conference on Artificial Intelligence*, volume 3339, pages 488–499. Springer, 2004.

[6] D. S. Mansjur, Q. Fu, and B. H. Juang. Utilizing non-uniform cost learning for active control of inter-class confusion. In *Proceedings of the 19th International Conference on Pattern Recognition,Tampa,FL,USA*, Dec. 2008.

[7] D. S. Mansjur and B. H. Juang. Improving kernel density classifier using corrective bandwidth learning with smooth error loss function. In *Proceedings of the 7th International Conference on Machine Learning and Applications, San Diego, CA, USA*, pages 161–167, Dec. 2008.

[8] B. Silverman. *Density Estimation for Statistics and Data Analysis*. Monographs on Statistics and Applied Probability. Chapman and Hall, London-New York, 1986.

[9] M. Wang, X.-S. Hua, Y. Song, X. Yuan, S. Li, and H.-J. Zhang. Automatic video annotation by semi-supervised learning with kernel density estimation. In *Proceedings of the 14th annual ACM international conference on Multimedia*, pages 967–976, New York, NY, USA, 2006.