Simultaneous Categorization of Text Documents and Identification of Cluster-dependent Keywords

Hichem Frigui and Olfa Nasraoui
Department of Electrical and Computer Engineering
University of Memphis
Campus Box 526574, Memphis, TN 38152
{hfrigui, onasraou}@memphis.edu

Abstract—

In this paper, we propose a new approach to unsupervised text document categorization based on a coupled process of clustering and cluster-dependent keyword weighting. The proposed algorithm is based on the K-Means clustering algorithm. Hence it is computationally and implementationally simple. Moreover, it learns a different set of keyword weights for each cluster. This means that, as a by-product of the clustering process, each document cluster will be characterized by a possibly different set of keywords. The cluster dependent keyword weights have two advantages. First, they help in partitioning the document collection into more meaningful categories. Second, they can be used to automatically generate a compact description of each cluster in terms of not only the attribute *values*, but also their *relevance*. In particular, for the case of *text* data, this approach can be used to automatically annotate the documents. The performance of the proposed algorithm is illustrated by using it to cluster a synthetic data set and a real collection of text documents.

I. Introduction

Clustering is an important task that is performed as part of many text mining and information retrieval systems. Clustering can be used for efficiently finding the nearest neighbors of a document [1], for improving the precision or recall in information retrieval systems [2], [3], for aid in browsing a collection of documents [4], and for the organization of search engine results [5], and lately for the personalization of search engine results [6].

Most current document clustering approaches work with what is known as the vector-space model, where each document is represented by a vector in the term-space. The latter generally consists of the keywords important to the document collection. For instance, the respective term or word frequencies (TF) [7]in a given document can be used to form a vector model for this document. In order to discount frequent words with little discriminating power, each term/word can be weighted based on its Inverse Document Frequency (IDF) [7], [6] in the document collection. However, the distribution of words in most real document collections can vary drastically from one group of documents to another. Hence relying solely on the IDF for keyword selection can be inappropriate and can severely degrade the results of clustering and/or any other learning tasks that follow it. For instance, a group of ”News” documents and a group of “Business” documents are expected to have different sets of important keywords. Now, if the documents have already been manually pre-classified into distinct categories, then it would be trivial to select a different set of keywords for each category based on IDF. However, for large dynamic document collections, such as the case of World Wide Web documents, this manual classification is impractical, hence the need for automatic or unsupervised classification/clustering that can handle categories that differ widely in their best keyword sets. Unfortunately, it is not possible to differentiate between different sets of keywords, unless the documents have already been categorized. This means that in an unsupervised mode, both the categories and their respective keyword sets need to be discovered *simultaneously*. Selecting and weighting subsets of keywords in text documents is smilar to the problem of feature selection and weighting in pattern recognition and data mining. The problem of selecting the best subset of features or attributes constitutes an important part of the design of good learning algorithms for real world tasks. Irrelevant features can significantly degrade the generalization performance of these algorithms. In fact, even if the data samples have already been classified into known classes, it is generally preferrable to model each complex class by several simple sub-classes or clusters, and to use a different set of feature weights for each cluster. This can help in classifying new documents into one of the pre-existing categories. So far, the problem of clustering and feature seletion have been treated rather independently or in a wrapper kind approach [8], [9], [10], [11], [12], [13], but rarely coupled together to achieve the same objective.

In [14], we have presented a new algorithm, called Simultaneous Clustering and Attribute Discrimination (SCAD), that performs clustering and feature weighting *simultaneously*. When used as part of a supervised or unsupervised learning system, SCAD offers several advantages. First, its *continuous* feature weighting provides a much richer feature relevance representation than binary feature selection. Secondly, SCAD learns a *different* feature relevance representation for each cluster in an *unsupervised* manner. However, SCAD was intended for use with data lying in some Euclidean space, and the distance measure used was the Euclidean distance. For the special case of text documents, it is well known that the Euclidean distance is not appropriate, and other measures such as the cosine similarity or Jackard index are better suited to assess the similarity/dissimilarity between documents.

In this paper, we extend SCAD to *simultaneous text* document clustering and *dynamic category-dependent* keyword set weighting. This new approach to text clustering, that we call "Simulatneous KeyWord Identification and Clustering of text documents" or *SKWIC*, is both conceptually and computationally simple, and offers the following advantages compared to existing document clustering techniques. First, its *continuous* term weighting provides a much richer feature relevance representation than binary feature selection: Not all terms are considered *equally* relevant in a *single* category of text documents. This is especially true when the number of keywords is large. For example, one would expect the word "playoff" to be more important than the word "program" to distinguish a group of "sports" documents. Secondly, a given term is not considered *equally* relevant in *all* categories: For instance, the word "film" may be more relevant to a group of "entertainment" related documents than to a group of "sports" documents. Finally, SKWIC *learns* a *different* set of term weights for each cluster in an *unsupervised* manner.

The organization of the rest of the paper is as follows. In section 2, we review the SCAD algorithm. In section 3, we modify SCAD for the case of text document categorization, and derive necessary conditions to update the term weights. In section 4, we illustrate the performance of SKWIC with synthetic and real examples. Finally, section 5 contains the summary conclusions.

II. Simultaneous Clustering and Attribute Discrimination

The Simultaneous Clustering and Attribute Discrimination (SCAD) algorithm [14] was designed to search for the optimal cluster centers, $\mathbf{C}$, and the optimal set of attribute weights, $\mathbf{V}$, simultaneously. Each cluster i is allowed to have its own set of feature weights $\mathbf{V}_i = [v_{i1}, \cdots, v_{in}]$ and fuzzy membership degrees $(u_{ij}$ that define a fuzzy partition of the data set. Without loss of generality, here we present the *crisp* case where memberships are binarized to 1 and 0 based on minimum distance fom a data point to a cluster prototype. We also note that crisper memberships may be preferrable for the case of clustering *text* documents because of the loss of information that can be caused by the distinct nature of text data, the dissimilarity measures involved, and excessive fuzziness in the presence of extremely large numbers of attributes. For the crisp case, SCAD attempts to minimize the following objective function:

$$\begin{aligned} J(\mathbf{C}, \mathbf{V}; \mathcal{X}) &= \sum_{i=1}^{C} \sum_{x_j \in \mathcal{X}_i} \sum_{k=1}^{n} v_{ik}(x_{jk} - c_{ik})^2 \\ &\quad + \sum_{i=1}^{C} \delta_i \sum_{k=1}^{n} v_{ik}^2, \end{aligned} \tag{1}$$

subject to

$$v_{ik} \in [0,1] \, \forall \, i, \, k; \quad \text{and} \quad \sum_{k=1}^{n} v_{ik} = 1, \, \forall \, i. \tag{2}$$

In (1), x_{jk} is the k^{th} feature value of $n-$dimensional data point $\mathbf{x}_j = [x_{j1}, \cdots, x_{jn}]$, c_{ik} is the k^{th} component of the i^{th} cluster center vector, and $\mathbf{V} = [v_{ik}]$ is the relevance weight of feature k in cluster i, and $\mathcal{X}_i$ is the set of data samples assigned to the i^{th} cluster. For the case of the *Euclidean* distance measure, it was shown that the optimal feature weights are given by [14]

$$v_{ik} = \frac{1}{n} + \frac{1}{2\delta_i} \sum_{x_j \in \mathcal{X}_i} \left[\frac{\|\mathbf{x}_j - \mathbf{c}_i\|^2}{n} - (x_{jk} - c_{ik})^2 \right]. \tag{3}$$

The first term in (3), $(1/n)$, is the default value if all attributes are treated equally, and no discrimination is performed. The second term is a bias that can be either positive or negative. It is positive for compact features where the distance along this dimension is, on the average, less than the total distance using all of the dimensions. If an attribute is very compact, compared to the other attributes, for most of the points that belong to a given cluster, then it is very relevant for that cluster.

The choice of δ_i in equation (1) is important in the SCAD algorithm since it reflects the importance of the second term relative to the first term. If δ_i is too small, then only one feature in cluster i will be relevant and assigned a weight of one. All other features will be assigned zero weights. On the other hand, if δ_i is too large, then all features in cluster i will be relevant, and assigned equal weights of $1/n$. The values of δ_i is chosen dynamically such that both terms are of the same order of magnitude [14] .

It can also be shown that the cluster partition that minimizes J is the one that assigns each data sample to the cluster with *nearest* prototype/center, i.e.,

$$\mathcal{X}_i = \left\{ \mathbf{x}_j | \tilde{d}_{ij}^2 \leq \tilde{d}_{kj}^2 \forall k \neq i \right\} \tag{4}$$

where

$$\tilde{d}_{ij}^2 = \sum_{k=1}^{n} v_{ik}(x_{jk} - c_{ik})^2 \tag{5}$$

is the weighted aggregate Euclidean distance, and ties are resolved arbitrarily.

Similarly, a mathematical optimization procedure was used in [14] to minimize J with respect to the centers, to obtain

$$c_{ik} = \begin{cases} 0 & \text{if } v_{ik} = 0, \\ \frac{\sum_{x_j \in \mathcal{X}_i} x_{jk}}{\sum_{x_j \in \mathcal{X}_i}} & \text{if } v_{ik} > 0 \end{cases} . \tag{6}$$

As expected, these center update equations are similar to those of K-Means, because the second term of the objective function in (1) is independent of the centers.

III. Simultaneous Clustering and Term Weighting of Text Documents

SCAD [14] was formulated based on Euclidean distance. However, for many data mining applications such as clustering *text* documents and other *high dimensional* data sets, the Euclidean distance measure is not appropriate. In general, the Euclidean distance are not good measures for document categorization. This is due mainly to the high dimensionality of the problem, and the fact that two documents may not be considered similar if keywords are missing in both documents. More appropriate for this application, is the cosine similarity measure, [7],

$$S(O_i, O_j) = \frac{\sum_{k=1}^{p} y_{ik} \times y_{jk}}{\sqrt{\sum_{k=1}^{p} y_{ik}^2}\sqrt{\sum_{k=1}^{p} y_{jk}^2}} \tag{7}$$

In order to be able to extend SCAD's criterion function for the case when another dissimilarity measure is employed, we only require the ability to decompose the dissimilarity measure across the different attribute directions. In this paper, we will attempt to decouple a dissimilarity based on the cosine similarity measure. We accomplish this by defining the dissimilarity between document $\mathbf{x}_j$ and the i^{th} cluster center vector as follows

$$\tilde{D}_{wc_{ij}} = \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^k, \tag{8}$$

which is the Weighted aggregate sum of Cosine-based distances along the individual dimensions, where

$$D_{wc_{ij}}^k = \frac{1}{n} - (x_{jk}.c_{ik}), \tag{9}$$

x_{jk} is the frequency of the k^{th} term in document $\mathbf{x}_j$, c_{ik} is the k^{th} component of the i^{th} cluster center vector, and $\mathbf{V} = [v_{ik}]$ is the relevance weight of keyword k in cluster i. Note that the individual products are not normalized in (8) because it is assumed that the data vectors are normalized to unit length before they are clustered, and that all cluster centers are normalized after they are updated in each iteration.

SKWIC is designed to search for the optimal cluster centers, $\mathbf{C}$, and the optimal set of feature weights, $\mathbf{V}$, simultaneously. Each cluster i is allowed to have its own set of feature weights $\mathbf{V}_i = [v_{i1}, \cdots, v_{in}]$. We define the following objective function:

$$\begin{aligned} J(\mathbf{C}, \mathbf{V}; \mathcal{X}) &= \sum_{i=1}^{C} \sum_{x_j \in \mathcal{X}_i} \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^k \\ &\quad + \sum_{i=1}^{C} \delta_i \sum_{k=1}^{n} v_{ik}^2, \end{aligned} \tag{10}$$

subject to

$$v_{ik} \in [0,1] \; \forall\, i,\ k; \quad \text{and} \quad \sum_{k=1}^{n} v_{ik} = 1,\ \forall\, i. \tag{11}$$

The objective function in (10) has two components. The first component, is the sum of distances or errors to the cluster centers. This component allows us to obtain compact clusters. It is minimized when only one keyword in each cluster is completely relevant, and all other keywords are irrelevant. The second component in equation (10) is the sum of the squared keyword weights. The global minimum of this component is achieved when all the keywords are equally weighted. When both components are combined and δ_i are chosen properly, the final partition will minimize the sum of intra-cluster weighted distances, where the keyword weights are optimized for each cluster.

To optimize J, with respect to $\mathbf{V}$, we use the Lagrange multiplier technique, and obtain

$$\begin{aligned} J(\mathbf{\Lambda}, \mathbf{V}) &= \sum_{i=1}^{C} \sum_{x_j \in \mathcal{X}_i} \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^k \\ &\quad + \sum_{i=1}^{C} \delta_i \sum_{k=1}^{n} v_{ik}^2 - \sum_{i=1}^{C} \lambda_i \Big(\sum_{k=1}^{n} v_{ik} - 1 \Big), \end{aligned}$$

where $\Lambda = [\lambda_1, \cdots, \lambda_c]^t$. Since the rows of $\mathbf{V}$ are independent of each other, we can reduce the above optimization problem to the following C independent problems:

$$\begin{aligned} J_i(\lambda_i, \mathbf{V}_i) &= \sum_{x_j \in \mathcal{X}_i} \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^k \\ &\quad + \delta_i \sum_{k=1}^{n} v_{ik}^2 - \lambda_i \Big(\sum_{k=1}^{n} v_{ik} - 1 \Big) \\ &\quad \text{for } i = 1, \cdots, C, \end{aligned}$$

where $\mathbf{V}_i$ is the i^{th} row of $\mathbf{V}$. By setting the gradient of J_i to zero, we obtain

$$\frac{\partial J_i(\lambda_i, \mathbf{V}_i)}{\partial \lambda_i} = \Big(\sum_{k=1}^{n} v_{ik} - 1 \Big) = 0, \qquad (12)$$

and

$$\frac{\partial J_i(\lambda_i, \mathbf{V}_i)}{\partial v_{ik}} = \sum_{x_j \in \mathcal{X}_i} D^k_{wc_{ij}} + 2\delta_i v_{ik} - \lambda_i = 0. \qquad (13)$$

Solving (12) and (13) for v_{ik}, we obtain

$$v_{ik} = \frac{1}{n} + \frac{1}{2\delta_i} \sum_{x_j \in \mathcal{X}_i} \Big[\frac{\tilde{D}_{wc_{ij}}}{n} - D^k_{wc_{ij}} \Big]. \qquad (14)$$

The first term in (14), $(1/n)$, is the default value if all attributes/keywords are treated equally, and no discrimination is performed. The second term is a bias that can be either positive or negative. It is positive for compact attributes where the distance along this dimension is, on the average, less than the total distance using all of the dimensions. If an attribute is very compact, compared to the other attributes, for most of the points that belong to a given cluster, then it is very relevant for that cluster. Note that it is possible for the individual term-wise dissimilarities in (9) to become negative. This will simply emphasize that dimension further and will result in relatively larger attribute weights v_{ik} (see (14)). Moreover, the total aggregate dissimilarity in (8) can become negative. This also does not pose a problem because we partition the data based on minimum distance.

The choice of δ_i in equation (10) is important in the SKWIC algorithm since it reflects the importance of the second term relative to the first term. If δ_i is too small, then only one keyword in cluster i will be relevant and assigned a weight of one. All other words will be assigned zero weights. On the other hand, if δ_i is too large, then all words in cluster i will be relevant, and assigned equal weights of $1/n$. The values of δ_i should be chosen such that both terms are of the same order of magnitude. In all examples described in this paper, we compute δ_i in iteration, t, using

$$\delta_i^{(t)} = K \frac{\sum_{x_j \in \mathcal{X}_i} \sum_{k=1}^{n} v_{ik}^{(t-1)} \big(D^{k^{(t-1)}}_{wc_{ij}} \big)}{\sum_{k=1}^{n} \big(v_{ik}^{(t-1)} \big)^2}. \qquad (15)$$

In (15), K is a constant, and the superscript $(t-1)$ is used on u_{ij}, v_{ik}, and c_{ik} to denote their values in iteration $(t-1)$.

It should be noted that depending on the values of δ_i, the feature relevance values v_{ik} may not be confined to [0,1]. If this occurs very often, then it is an indication that the value of δ is too small, and that it should be increased (increase K). On the other hand, if this occurs for few clusters and only in few iterations, then it is safe to simply set negative values to zero, and to clip values that are greater than one to one.

It can also be shown that the cluster partition that minimizes J is the one that assigns each data sample to the cluster with *nearest* prototype/center, i.e.,

$$\mathcal{X}_i = \Big\{ \mathbf{x}_j | \tilde{D}_{wc_{ij}} \le \tilde{D}_{wc_{kj}} \forall k \ne i \Big\} \qquad (16)$$

where $\tilde{D}_{wc_{kj}}$ is the weighted aggregate cosine based distance in (8), and ties are resolved arbitrarily.

It is not possible to minimize J with respect to the centers. Hence, we will compute the new cluster centroids (as in the ordinary SCAD algorithm [14]) and normalize them to unit length to obtain the new cluster center. We obtain two cases depending on the value of v_{ik}.

Case 1: $v_{ik} = 0$

In this case the k^{th} feature is completely irrelevant relative to the i^{th} cluster. Hence, regardless of the value of c_{ik}, the values of this feature will not contribute to the overall weighted distance computation. Therefore, in this situation, any arbitrary value can be chosen for c_{ik}. In practice, we set $c_{ik} = 0$.

Case 2: $v_{ik} \neq 0$

For the case when the k^{th} feature has some relevance to the i^{th} cluster, equation (**??**) reduces to

$$c_{ik} = \frac{\sum_{x_j \in \mathcal{X}_i} x_{jk}}{\sum_{x_j \in \mathcal{X}_i}}.$$

To summarize, the update equation for the centers is

$$c_{ik} = \begin{cases} 0 & \text{if } v_{ik} = 0, \\ \frac{\sum_{x_j \in \mathcal{X}_i} x_{jk}}{\sum_{x_j \in \mathcal{X}_i}} & \text{if } v_{ik} > 0 \end{cases} \qquad (17)$$

Finally, we summarize the SKWIC algorithm below.

Simultaneous Keyword Identification and Clustering of text documents (SKWIC)

```
Fix the number of clusters $C$;
Initialize the centers by randomly selecting $C$ documents;
Initialize the partitions, $\mathcal{X}_i$, using (16) and equal feature weights ($\frac{1}{n}$);
REPEAT
  Compute $D^k_{wc_{ij}} = \frac{1}{n} - (x_{jk}.c_{ik})$
    for $1 \le i \le C$, $1 \le j \le N$, and $1 \le k \le n$;
  Update the relevance weights $v_{ik}$ by using (14);
  Update the cluster partition $\mathcal{X}_i$ by using (16);
  Update the centers by using (17);
  Update $\delta_i$ by using (15);
UNTIL ( centers stabilize );
```

The feature weighting equations used in SKWIC may be likened to the estimation and use of a covariance matrix in an inner-product norm-induced metric [15] in various statistical pattern recognition techniques. However, the estimation of a covariance matrix does not really weight the attributes according to their relevance, and it relies on the assumption that the data has a multivariate Gaussian distribution. On the other hand, SKWIC is free of any such assumptions when estimating the feature weights. This means that SKWIC can be adapted to more general dissimilarity measures, such as was done inthis paper with the cosine-based dissimilarity.

IV. Experimental Results

First, we present the results of the SCAD algorithm to illustrate the need for simultaneous clustering and feature discrimination. We generate two synthetic Gaussian clusters with the following mean vectors and covariance matrices: $(\mu_1, \sum_1) = ([0,0]^T, \mathbf{I}_2)$ and $(\mu_2, \sum_2) = ([5,5]^T, \mathbf{I}_2)$. Each cluster contains 20 points. Table I shows the results when K-Means is used to cluster this data set. The results obtained using SCAD are displayed in Table III. Since both features are almost equally relevant for both clusters, they have high weights (0.46 and 0.54 in both clusters), and the centers are close to the actual centers.

To demonstrate the ability of the proposed algorithm to cluster and identify relevant features, we increase the number of features to four by adding two irrelevant features to each cluster. These additional features are shown in bold in Table V. The first two features of the first cluster are uniformly distributed in the intervals $[0, 20]$ and $[0, 10]$ respectively. Features two and four of the second cluster are uniformly distributed in the intervals $[0, 10]$ and $[0, 5]$ respectively. A traditional feature selection algorithm can only discriminate against the second feature since it is irrelevant for both clusters. Clustering the remaining three features will not provide a compact description of each cluster.

TABLE I
RESULTS OF K-MEANS ON TWO GAUSSIAN CLUSTERS WITH ONLY RELEVANT FEATURES

	Cluster # 1		**Cluster # 2**	
Features	x_1	x_2	x_1	x_2
Centers	-0.36	0.28	4.64	5.28

SCAD converged after 10 iterations, and the results are displayed in Table VIII. The first feature of the first cluster is correctly identified as irrelevant ($v_{11} = 0.0$). The second feature has a higher weight ($v_{12} = 0.16$) because it has a relatively smaller dynamic range. Feature four of the second cluster was not identified as completely irrelevant ($v_{23} = 0.26$). This is because it has a dynamic range

TABLE II
RESULTS OF K-MEANS ON TWO GAUSSIAN CLUSTERS WITH ONLY RELEVANT FEATURES

	Cluster # 1	**Cluster # 2**
# of correctly labeled samples	20	20
# of incorrectly labeled samples	0	0

TABLE III
RESULTS OF SCAD ON TWO GAUSSIAN CLUSTERS WITH ONLY RELEVANT FEATURES

	Cluster # 1		**Cluster # 2**	
Features	x_1	x_2	x_1	x_2
Centers	-0.36	0.28	4.64	5.28
Relevance Weights	0.46	0.54	0.46	0.54

TABLE IV
RESULTS OF SCAD ON TWO GAUSSIAN CLUSTERS WITH ONLY RELEVANT FEATURES

	Cluster # 1	**Cluster # 2**
# of correctly labeled samples	20	20
# of incorrectly labeled samples	0	0

TABLE V
TWO 4-DIMENSIONAL CLUSTERS (ADDED CLUSTER-DEPENDENT FEATURES ARE HIGHLIGHTED)

Cluster # 1				**Cluster # 2**			
x_1	x_2	x_3	x_4	x_1	x_2	x_3	x_4
19.00	**2.09**	-0.33	1.11	4.66	**2.13**	6.11	**0.28**
4.62	**3.79**	-2.02	-0.73	2.97	**6.43**	4.26	**1.76**
12.13	**7.83**	-0.33	0.72	4.66	**3.20**	5.72	**4.06**
9.71	**6.80**	-0.25	0.04	4.74	**9.60**	5.04	**0.04**
17.82	**4.61**	-1.08	-0.37	3.91	**7.26**	4.62	**0.69**
15.24	**5.67**	0.15	-0.36	5.15	**4.11**	4.63	**1.01**
9.12	**7.94**	-1.22	0.11	3.77	**7.44**	5.11	**0.99**
0.37	**0.59**	1.80	1.43	6.80	**2.67**	6.43	**3.01**
16.42	**6.02**	-1.48	-0.70	3.51	**4.39**	4.29	**1.36**
8.89	**0.50**	-0.87	1.02	4.12	**9.33**	6.02	**0.99**
12.30	**4.15**	-0.21	-0.45	4.78	**6.83**	4.54	**0.07**
15.83	**3.05**	-0.28	1.06	4.71	**2.12**	6.06	**3.73**
18.43	**8.74**	0.45	0.16	5.45	**8.39**	5.16	**2.22**
14.76	**0.15**	-2.29	1.98	2.74	**6.28**	6.98	**4.65**
3.52	**4.98**	0.84	-0.68	5.84	**1.33**	4.31	**2.33**
8.11	**7.67**	1.49	1.61	6.49	**2.07**	6.61	**2.09**
18.70	**9.70**	-0.23	0.31	4.76	**6.07**	5.31	**4.23**
18.33	**9.90**	-0.46	-0.82	4.53	**6.29**	4.17	**2.62**
8.20	**7.88**	-1.58	-1.09	3.41	**3.70**	3.90	**1.01**
17.87	**4.38**	0.72	1.27	5.72	**5.75**	6.27	**3.36**

TABLE VI
RESULTS OF K-MEANS ON THE DATA SET WITH CLUSTER-DEPENDENT IRRELEVANT FEATURES IN TABLE V

	Cluster # 1				**Cluster # 2**			
Features	x_1	x_2	x_3	x_4	x_1	x_2	x_3	x_4
Centers	**16.41**	**5.53**	-0.45	0.33	5.19	**5.20**	3.71	**3.70**

TABLE VII

RESULTS OF K-MEANS ON TWO GAUSSIAN CLUSTERS WITH CLUSTER-DEPENDENT IRRELEVANT FEATURES

	Cluster # 1	**Cluster # 2**
# of correctly labeled samples	12	20
# of incorrectly labeled samples	8	0

TABLE VIII

RESULTS OF SCAD ON THE DATA SET WITH CLUSTER-DEPENDENT IRRELEVANT FEATURES IN TABLE V

	Cluster # 1				**Cluster # 2**			
Features	x_1	x_2	x_3	x_4	x_1	x_2	x_3	x_4
Centers	**12.47**	**5.33**	-0.36	0.28	4.64	**5.27**	5.28	**2.03**
Relevance Weights	**0.00**	**0.16**	0.42	0.42	0.35	**0.00**	0.39	**0.26**

TABLE IX

RESULTS OF SCAD ON TWO GAUSSIAN CLUSTERS WITH CLUSTER-DEPENDENT IRRELEVANT FEATURES

	Cluster # 1	**Cluster # 2**
# of correctly labeled samples	20	20
# of incorrectly labeled samples	0	0

close to the actual features, and therefore it will be treated as almost equally important.

The next experiment illustates the clustering results on a collection of text documents collected from the World Wide Web. Students were asked to collect 50 distinct documents from each of the following categories: news, business, entertainment, and sports. Thus the entire collection consists of 200 documents. The documents contents were preprocessed by eliminating stop words and stemming words to their root source. Then the Inverse Document Frequencies (IDF) [7] of the terms were computed and sorted in descending order so that only the top 200 terms were chosen as final keywords. Finally each document was represented by the vector of its document frequencies, and this vector was normalized to unit length. Using $C = 4$ as the number of clusters, SKWIC converged after 5 iterations, resulting in a partition that closesly resembles the distribution of the documents with respect to their respective categories. Moreover, the collection of terms receiving highest feature relevance weights in each cluster reflected the general topic of the category winning the majority of the documents that were assigned to the cluster by SKWIC. We show for each cluster, only six of the words with relevance weight $v_{ik} \geq \frac{1}{n} = \frac{1}{200} = 0.005$. The class distribution is shown in Table X. Class 2 showed most of the error in assignment due to the mixed nature of some of the documents therein. For example, by looking at the excerpts (shown below) from the following documents from class 2 (*entertainment*) that were assigned to cluster 1 with relevant words relating to *business* as seen in Table XI, one can see that these documents are hard to classify into one category, and that the keywords present in the documents in this case have mislead the clustering process.

Excerpt from Document 54: *... The couple were together for 3-1/2 years before their highly publicized split last month. Now, their Ojai property is on the market for $2.75 million, the Los Angeles Times reported on Sunday.*

The pair bought the 10-acre Ojai property – complete with working avocado and citrus orchards – at the end of 1998. They also purchased a Hollywood Hills home for $1.7 million in June 1999, according to the Times....

Excerpt from Document 59:

... The recommendation, approved last week by the joint strike committee for the Screen Actors Guild (SAG) and the American Federation of Television & Radio Artists (AFTRA), would have to be approved by the national boards of the unions to go into effect – a process that would take a month to complete.

"Part of this is motivated by the awareness of actors who have been egregious about performing struck work and part of it is trying to recognize the 99.999% of members who have stuck together on this," SAG spokesman Greg Krizman said...

Excerpt from Document 78:

... The Oxford-based quintet's acclaimed fourth release, "Kid A," opened at No. 1 with sales of 207,000 copies in the week ended Oct. 8, the group's Capitol Records label said Wednesday. The tally is more than four times the first-week sales of its previous album.

The last Stateside No. 1 album from the U.K was techno act Prodigy's "The Fat of the Land" in July 1997. That very same week, Radiohead's "OK Computer" opened at No. 21 with 51,000 units sold. It went on to sell 1.2 million copies in the United States...

Finally, we note that relevant keywords such as shown in Table XI can be used to provide a *short summary* for each cluster and to automatically *annotate* documents.

TABLE X

DISTRIBUTION OF THE 50 DOCUMENTS FROM EACH CLASS INTO THE 4 CLUSTERS

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**
	(business)	**(entertainment)**	**(news)**	**(sports)**
class 1	45	2	3	0
class 2	9	31	4	6
class 3	1	1	47	1
class 4	0	0	4	46

V. CONCLUSION

In this paper, we presented a new approach that performs clustering and attribute weighting simultaneously

TABLE XI

TERM RELEVANCE FOR THE TOP SIX RELEVANT WORDS IN EACH CLUSTER

Cluster # 1		Cluster # 2		Cluster # 3		Cluster # 4	
$v_{1(k)}$	$w_{(k)}$	$v_{2(k)}$	$w_{(k)}$	$v_{3(k)}$	$w_{(k)}$	$v_{4(k)}$	$w_{(k)}$
0.028	compani	0.031	film	0.009	polic	0.021	game
0.015	percent	0.012	star	0.008	nation	0.013	season
0.010	share	0.010	dai	0.008	state	0.012	open
0.010	expect	0.010	week	0.008	offici	0.009	york
0.009	market	0.009	peopl	0.008	sai	0.008	hit
0.008	stock	0.008	like	0.007	kill	0.008	run

and in an unsupervised manner. Our approach is an extension of the K-Means algorithm, that in addition to partitionning the data set into a given number of clusters, also finds an optimal set of feature weights for *each* cluster. SKWIC minimizes one objective function for both the optimal prototype parameters and feature weights for each cluster. This optimization is done iteratively by dynamically updating the prototype parameters and the attribute relevance weights in each iteration. This makes the proposed algorithm simple and fast.

Our experimental results showed that SKWIC's performance is comparable to K-Means when all the attributes are equally important for all clusters. However, SKWIC outperforms K-Means when not all the features are equally relevant to all clusters. This makes our approach more reliable, especially, when clustering in *high dimensional* spaces, as in the case of categorization of text documents, where not all attributes are equally important, and where clusters tend to form in only *subspaces* of the original feature space. Also, for the case of *text* data, this approach can be used to automatically annotate the documents.

Since the objective function of SKWIC is based on that of the K-Means, it inherits most of the advantages of K Mean-type clustering algorithms, such as ease of computation and simplicity. Moreover, because K-Means has been studied extensively over the last decades, the proposed approach can easily benefit from the advances and improvements that led to several K-Means variants in the data mining and pattern recognition communities. In particular, the techniques developed to handle noise [16], to determine the number of clusters [17], to cluster very large data sets [18], [19], and to improve initialization [20]. We are currently investigating these extensions.

REFERENCES

[1] C. Buckley and A.F. Lewit, “Optimizations of inverted vector searches,” in *SIGIR '85*, 1985, pp. 97–110.

[2] C.J. VanRijsbergen, *Information Retrieval*, Buttersworth, London, 1989.

[3] G. Kowalski, *Information retrieval systems-theory and implementations*, Kluwer Academic Publishers, 1997.

[4] D.R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey, “Scatter/gather: A cluster-based approach to browsing large document collections,” in *SIGIR '92*, 1992, pp. 318–329.

[5] O. Zamir, O. Etzioni, O. Madani, and R.M. Karp, “Fast and intuitive clustering of web documents,” in *KDD'97*, 1997, pp. 287–290.

[6] D. Mladenic, “Text learning and related intelligent agents,” *IEEE Expert*, Jul. 1999.

[7] R. R. Korfhage, *Information Storage and Retrieval*, Wiley, 1997.

[8] H. Almuallim and T. G. Dietterich, “Learning with many irrelevant features,” in *Ninth National Conference on artificial intelligence*, 1991, pp. 547–552.

[9] K. Kira and L. A. Rendell, “The feature selection problem: Traditional methods and a new algorithm,” in *Tenth National Conference on artificial intelligence*, 1992, pp. 129–134.

[10] L. A. Rendell and K. Kira, “A practical approach to feature selection,” in *International Conference on machine learning*, 1992, pp. 249–256.

[11] G. John, R. Kohavi, and K. Pfleger, “Irrelevant features and the subset selection problem,” in *Eleventh International Machine Learning Conference*, 1994, pp. 121–129.

[12] D. Skalak, “Prototype and feature selection by sampling and random mutation hill climbing algorithms,” in *Eleventh International Machine Learning Conference (ICML-94)*, 1994, pp. 293–301.

[13] R. Kohavi and D. Sommerfield, “Feature subset selection using the wrapper model: Overfitting and dynamic search space topology,” in *First International Conference on Knowledge Discovery and Data Mining*, 1995, pp. 192–197.

[14] H. Frigui and O. Nasraoui, “Simultaneous clustering and attribute discrimination,” in *IEEE Conference on Fuzzy Systems*, San Antonio, Texas, 2000, pp. 158–163.

[15] E. E. Gustafson and W. C. Kessel, “Fuzzy clustering with a fuzzy covariance matrix,” in *IEEE CDC*, San Diego, California, 1979, pp. 761–766.

[16] H. Frigui and R. Krishnapuram, “A robust competitive clustering algorithm with applications in computer vision,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 450–465, May. 1999.

[17] H. Frigui and R. Krishnapuram, “Clustering by competitive agglomeration,” *Pattern Recognition*, vol. 30, no. 7, pp. 1223–1232, 1997.

[18] P. S. Bradley, Usama M. Fayyad, and Cory Reina, “Scaling clustering algorithms to large databases,” in *Knowledge Discovery and Data Mining*, 1998, pp. 9–15.

[19] Fredrik Farnstrom, James Lewis, and Charles Elkan, “Scalability for clustering algorithms revisited,” *SIGKDD Explorations*, vol. 2, no. 1, pp. 51–57, 2000.

[20] Paul S. Bradley and Usama M. Fayyad, “Refining initial points for K-Means clustering,” in *Proc. 15th International Conf. on Machine Learning*. 1998, pp. 91–99, Morgan Kaufmann, San Francisco, CA.