# Simultaneous Clustering and Dynamic Keyword Weighting for Text Documents

Hichem Frigui
Olfa Nasraoui

ABSTRACT
In this chapter, 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. We also extend the proposed approach to handle the inherent fuzziness in text documents, by automatically generating fuzzy or soft labels instead of hard all-or-nothing categorization. This means that a text document can belong to *several* categories with different degrees. The proposed approach can handle noise documents elegantly by automatically designating one or two *noise magnet* clusters that grab most outliers away from the other clusters. The performance of the proposed algorithm is illustrated by using it to cluster real text document collections.

## 1 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 [BL85], for improving the precision or recall in information retrieval systems [Van89, Kow97], for aid in browsing a collection of documents [CKPT92], and for the organization of search engine results [ZEMK97], and lately for the personalization of search engine results [Mla99].

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 Frequencies (TF) [Kor97] 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) [Kor97, Mla99] 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 [AD91, KR92, RK92, JKP94, Ska94, KS95], but rarely coupled together to achieve the same objective.

In [FN00], 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 chapter, 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.

We also extend the proposed approach to handle the inherent fuzziness in text documents, by automatically generating fuzzy or soft labels instead of single-label categorization. This means that a text document can belong to *several* categories with different degrees.

By virtue of the dynamic keyword weighting, and its continuous interaction with distance and membership computations, the proposed approach is able to handle noise documents elegantly by automatically designating one or two *noise magnet* clusters that grab most outliers away from the other clusters.

The organization of the rest of the chapter is as follows. In section 2, we present the criterion for "Simulatneous KeyWord Identification and Clustering of text documents" or *SKWIC*, and derive necessary conditions to update the term weights. In Section 3, we present an alternative clustering technique, *Fuzzy SKWIC*, that provides richer *soft* document partitions. In Section 4, we explain how our approach achieves *robustness* to outliers in the data set. In section 5, we illustrate the performance of SKWIC in unsupervised categorization of several text collections. Finally, section 6 contains the summary conclusions.

## 2 Simultaneous Clustering and Term Weighting of Text Documents

SCAD [FN00] 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 is not a good measure 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, [Kor97],

$$S(O_i, O_j) = \frac{\sum_{k=1}^{p} y_{ik} \times y_{jk}}{\sqrt{\sum_{k=1}^{p} y_{ik}^2}\sqrt{\sum_{k=1}^{p} y_{jk}^2}} \tag{1.1}$$

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 work, 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

$$\tilde{D}_{wc_{ij}} = \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^{k}, \tag{1.2}$$

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

$$D_{wc_{ij}}^{k} = \frac{1}{n} - (x_{jk}.c_{ik}), \tag{1.3}$$

$x_{jk}$ 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 (1.2) 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:

$$\begin{aligned} J(\mathbf{C}, \mathbf{V}; \mathcal{X}) &= \sum_{i=1}^{C} \sum_{x_j \in \mathcal{X}_i} \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^{k} \\ &\quad + \sum_{i=1}^{C} \delta_i \sum_{k=1}^{n} v_{ik}^2, \end{aligned} \tag{1.4}$$

subject to

$$v_{ik} \in [0,1] \; \forall \; i, \; k; \quad \text{and} \quad \sum_{k=1}^{n} v_{ik} = 1, \; \forall \; i. \tag{1.5}$$

The objective function in (1.4) 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 (1.4) 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 $\delta_i$ 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

$$\frac{\partial J_i(\lambda_i, \mathbf{V}_i)}{\partial \lambda_i} = \Big( \sum_{k=1}^{n} v_{ik} - 1 \Big) = 0, \qquad (1.6)$$

and

$$\frac{\partial J_i(\lambda_i, \mathbf{V}_i)}{\partial v_{ik}} = \sum_{x_j \in \mathcal{X}_i} D_{wc_{ij}}^k + 2\delta_i v_{ik} - \lambda_i = 0. \qquad (1.7)$$

Solving (1.6) and (1.7) for $v_{ik}$, we obtain

$$v_{ik} = \frac{1}{n} + \frac{1}{2\delta_i} \sum_{x_j \in \mathcal{X}_i} \Big[ \frac{\sum_{x_j \in \mathcal{X}_i} D_{wc_{ij}}^k}{n} - D_{wc_{ij}}^k \Big]. \qquad (1.8)$$

The first term in (1.8), $(1/n)$, 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 (1.3) to become negative. This will simply emphasize that dimension further and will result in relatively larger attribute weights $v_{ik}$ (see (1.8)). Moreover, the total aggregate dissimilarity in (1.2) can become negative. This also does not pose a problem because we partition the data based on minimum distance.

The choice of $\delta_i$ in equation (1.4) is important in the SKWIC algorithm since it reflects the importance of the second term relative to the first term. If $\delta_i$ 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 $\delta_i$ is too large, then all words in cluster $i$ will be relevant, and assigned equal weights of $1/n$. The values of $\delta_i$ should be chosen such that both terms are of the same order of magnitude. In all examples described in this chapter, we compute $\delta_i$ in iteration, $t$, using

$$\delta_i^{(t)} = K_\delta \frac{\sum_{x_j \in \mathcal{X}_i} \sum_{k=1}^{n} v_{ik}^{(t-1)} \left( D_{wc_{ij}}^{k^{(t-1)}} \right)}{\sum_{k=1}^{n} \left( v_{ik}^{(t-1)} \right)^2}. \tag{1.9}$$

In (1.9), $K_\delta$ 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 $\delta_i$, 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 $\delta$ is too small, and that it should be increased (increase $K_\delta$). On the other hand, if this occurs for a few clusters and only in a few iterations, then we adjust the negative feature relevance values as follows:

$$v_{ik} \leftarrow v_{ik} + \left| \min_{k=1}^{n} v_{ik} \right| \quad \text{if } v_{ik} < 0 \tag{1.10}$$

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.,

$$\mathcal{X}_i = \left\{ \mathbf{x}_j | \tilde{D}_{wc_{ij}} \leq \tilde{D}_{wc_{kj}} \forall k \neq i \right\} \tag{1.11}$$

where $\tilde{D}_{wc_{kj}}$ is the weighted aggregate cosine based distance in (1.2), 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 [FN00]) and normalize them to unit length to obtain the new cluster centers. 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, the center 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

$$c_{ik} = \begin{cases} 0 & \text{if } v_{ik} = 0, \\ \frac{\sum_{x_j \in \mathcal{X}_i} x_{jk}}{|\mathcal{X}_i|} & \text{if } v_{ik} > 0 \end{cases} \tag{1.12}$$

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 (1.11) 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 (1.8);
  Compute $\tilde{D}_{wc_{ij}}$ for $1 \le i \le C$, $1 \le j \le N$, using (1.2);
  Update the cluster partition $\mathcal{X}_i$ by using (1.11);
  Update the centers by using (1.12);
  Update $\delta_i$ by using (1.9);
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 [GK79] 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 in this chapter with the cosine-based dissimilarity.

## 3 Simultaneous Soft Clustering and Term Weighting of Text Documents

Documents in a collection can rarely be described as members of a single/exclusive category. In fact most documents will tend to straddle in their subject between two or more different subjects. Even manual classification is difficult and poor in this case, because each document is finally labeled into a single class, and this can drastically affect retrieval abilities once a classification model is built. Hard partitioning models such as K-Means and SKWIC are constrained to assign every document to a single cluster/category, and the final assignment is often poor in modeling documents that can be assigned to more than one category. Consequently they are expected to have limited capability for real large document collections. In this section, we present a technique to provide a *soft* unsupervised categorization of a collection of documents. By soft, it is meant that a given document must not be confined to a single category.

It is known that for complex data sets containing overlapping clusters, fuzzy/soft partitions model the data better than their crisp/hard counterparts. In particular, fuzzy memberships are richer than crisp memberships in describing the degrees of belongingness of data points lying in the areas of overlap. Moreover, fuzzy partitions generally smoothen the surface of the criterion function in the search space, and hence, make the optimization process less prone to local or sub-optimal solutions. With a fuzzy partition, a data point $\mathbf{x}_j$ belongs to each cluster, $\mathcal{X}_i$, to a varying degree called fuzzy membership $u_{ij}$. A fuzzy partition, usually represented by the $C \times N$ matrix $\mathbf{U} = [u_{ij}]$ is called a constrained fuzzy $C-$partition of $\mathcal{X}$ if the entries of $\mathbf{U}$ satisfy the following constraints [Bez81],

$$\begin{cases} u_{ij} \in [0,1] & \forall i \\ 0 < \sum_{j=1}^{N} u_{ij} < N & \forall i,j \\ \sum_{i=1}^{C} u_{ij} = 1 & \forall j. \end{cases} \tag{1.13}$$

Fuzzy-SKWIC is designed to search for the optimal cluster centers, $\mathbf{C}$, the optimal soft partitioning memberships, $\mathbf{U}$, and the optimal set of feature weights, $\mathbf{V}$, simultaneously. Each cluster $i$ is allowed to have its own set of feature weights $\mathbf{V}_i = [v_{i1}, \cdots, v_{in}]$, and fuzzy membership degrees ($u_{ij}$ that define a fuzzy partition of the data set satisfying (1.13). We define the following objective function:

$$\begin{aligned} J(\mathbf{C},\mathbf{U},\mathbf{V};\mathcal{X}) &= \sum_{i=1}^{C}\sum_{j=1}^{N}(u_{ij})^m \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^k \\ &\quad + \sum_{i=1}^{C}\delta_i \sum_{k=1}^{n} v_{ik}^2, \end{aligned} \tag{1.14}$$

subject to

$$v_{ik} \in [0,1]\ \forall\ i,\ k; \quad \text{and} \quad \sum_{k=1}^{n} v_{ik} = 1,\ \forall\ i. \tag{1.15}$$

The objective function in (1.14) 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 (1.14) 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 $\delta_i$ 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

$$\begin{aligned} J(\mathbf{\Lambda},\mathbf{V}) &= \sum_{i=1}^{C}\sum_{j=1}^{N}(u_{ij})^m \sum_{k=1}^{n} v_{ik} D_{wc_{ij}}^k \\ &\quad + \sum_{i=1}^{C}\delta_i \sum_{k=1}^{n} v_{ik}^2 - \sum_{i=1}^{C}\lambda_i\Big(\sum_{k=1}^{n} v_{ik} - 1\Big), \end{aligned}$$

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:

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

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

$$\frac{\partial J_i(\lambda_i,\mathbf{V}_i)}{\partial \lambda_i} = \Big(\sum_{k=1}^{n} v_{ik} - 1\Big) = 0, \tag{1.16}$$

and

$$\frac{\partial J_i(\lambda_i, \mathbf{V}_i)}{\partial v_{ik}} = \sum_{j=1}^{N} (u_{ij})^m D_{wc_{ij}}^k + 2\delta_i v_{ik} - \lambda_i = 0. \tag{1.17}$$

Solving (1.16) and (1.17) for $v_{ik}$, we obtain

$$v_{ik} = \frac{1}{n} + \frac{1}{2\delta_i} \sum_{j=1}^{N} (u_{ij})^m \left[ \frac{\tilde{D}_{wc_{ij}}}{n} - D_{wc_{ij}}^k \right]. \tag{1.18}$$

The first term in (1.18), $(1/n)$, 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 (high $u_{ij}$), then it is very relevant for that cluster. Note that it is possible for the individual term-wise dissimilarities in (1.3) to become negative. This will simply emphasize that dimension further and will result in relatively larger attribute weights $v_{ik}$ (see (1.18)).

The choice of $\delta_i$ in equation (1.14) is important in the Fuzzy-SKWIC algorithm since it reflects the importance of the second term relative to the first term. If $\delta_i$ 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 $\delta_i$ is too large, then all words in cluster $i$ will be relevant, and assigned equal weights of $1/n$. The values of $\delta_i$ should be chosen such that both terms are of the same order of magnitude. In all examples described in this chapter, we compute $\delta_i$ in iteration, $t$, using

$$\delta_i^{(t)} = K_\delta \frac{\sum_{j=1}^{N} \left(u_{ij}^{(t-1)}\right)^m \sum_{k=1}^{n} v_{ik}^{(t-1)} \left(D_{wc_{ij}}^{k^{(t-1)}}\right)}{\sum_{k=1}^{n} \left(v_{ik}^{(t-1)}\right)^2}. \tag{1.19}$$

In (1.19), $K_\delta$ 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 $\delta_i$, 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 $\delta$ is too small, and that it should be increased (increase $K_\delta$). On the other hand, if this occurs for few clusters and only in few iterations, then we adjust the negative feature relevance values as follows:

$$v_{ik} \leftarrow v_{ik} + \left| \min_{k=1}^{n} v_{ik} \right| \text{ if } v_{ik} < 0 \tag{1.20}$$

Since the second term in (1.14) does not depend on $u_{ij}$ explicitly, the update equation of the memberships is similar to that of the Fuzzy C Means, i.e.,

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left( \frac{\tilde{D}_{wc_{ij}}}{\tilde{D}_{wc_{kj}}} \right)^{\frac{1}{m-1}}}. \tag{1.21}$$

The component wise distance values $D^k_{wc_{ij}}$ in (1.3) can be negative, and hence the overall distance $D^k_{wc_{ij}}$ in (1.2) can become negative, which can affect the sign of the fuzzy memberships in (1.21). Hence we adjust the negative distance values as follows:

$$\tilde{D}_{wc_{ij}} \leftarrow \tilde{D}_{wc_{ij}} + \left| \min_{i=1}^{C} \tilde{D}_{wc_{ij}} \right| \quad \text{if } \tilde{D}_{wc_{ij}} < 0 \tag{1.22}$$

Finally, the update equation for the centers which take into account the soft memberships/partition is

$$c_{ik} = \begin{cases} 0 & \text{if } v_{ik} = 0, \\ \frac{\sum_{j=1}^{N} (u_{ij})^m x_{jk}}{\sum_{j=1}^{N} (u_{ij})^m} & \text{if } v_{ik} > 0 \end{cases} \tag{1.23}$$

Finally, we summarize the Fuzzy-SKWIC algorithm below.

**Simultaneous Keyword Identification and**
**Clustering of text documents (Fuzzy-SKWIC)**

```
Fix the number of clusters $C$;
Fix $m$, $m \in [1, \infty)$;
Initialize the centers by randomly selecting $C$ documents;
Initialize the fuzzy partition matrix $\mathbf{U}$ ;
REPEAT
  Compute $D^k_{wc_{ij}} = \frac{1}{n} - (x_{jk}.c_{ik})$
   for $1 \leq i \leq C$, $1 \leq j \leq N$, and $1 \leq k \leq n$;
  Update the relevance weights $v_{ik}$ by using (1.18);
  Adjust relevance weights $v_{ik}$ by using (1.20);
  Compute $\tilde{D}_{wc_{ij}}$ for $1 \leq i \leq C$, $1 \leq j \leq N$, using (1.2);
  Adjust negative $\tilde{D}_{wc_{ij}}$ using (1.22);
  Update the partition matrix $\mathbf{U}^{(k)}$ by using (1.21);
  Update the centers by using (1.23);
  Update $\delta_i$ by using (1.19);
UNTIL ( centers stabilize );
```

# 4 Robustness in the Presence of Noise Documents

When there are several documents that do not form a strong consencus or cluster, i.e., they are neither similar to each other nor to any of the other compact clusters. Because our distance are confined in $[0, 1]$, all outlier documents will have a maximal distance of 1. Hence, their effect on the objective functions in (1.4) and (1.14) is limited. This means that they cannot drastically influence the results for other clusters. This limited influence, by definition, makes our approach *robust* in the face of outliers and noise. In essence, this is similar to using a $\rho()$ function in M-Estimators [Hub81, RL87].

Moreover, because the distance between ouliers and all clusters is close to the maximal value of 1, if they happen to get assigned to any one of the clusters initialized with a seed that is close to the outliers, they will tend to pull all the keyword relevance weights to a low value in that cluster because of extreme averaging. This in turn will further bias the distance computations to this cluster to be small. As a result, this cluster will start acting like a magnet that continues to grab documents that are not very typical of any category towards it, and therefore keep growing. Only documents that are really similar to their cluster's centroid will remain in their own clusters, and hence avoid to be pulled into the noise cluster. Consequently, designated *noise magnet* clusters will help in keeping the remaining clusters cleaner and their constituents more uniform.

We have observed the emergence of such *noise magnets* in every experiment that we performed.

# 5 Experimental Results

## *5.1 Simulation Results on 4-class Web Text Data*

### Simulation Results with Hard Clustering

The first 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) [Kor97] 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 closely resembles the distribution of the documents with respect to their true categories. The class distribution is shown in Table 1.1. Table 1.2 lists the six most relevant keywords for each cluster. As can be seen, 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. In addition, these cluster-dependent keywords can be used to provide a short summary for each cluster and to automatically annotate documents.

The partition of the documents of 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 1.2, 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...*

The above excerpts further illustrate the inherent *fuzziness* in categorizing text documents, as the shown documents straddle between the *business* and *entertainment* categories. In this case, it can be said that the baseline manual labeling was not accurate. Fuzzy or soft labels are desired for such documents, and these are illustrated in the next section.

TABLE 1.1. Distribution of the 50 documents from each class into the 4 clusters computed by SKWIC

| | **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** |

TABLE 1.2. Term relevance for the top six relevant words in each cluster computed by SKWIC

| **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 |

### Simulation Results with Soft Clustering

Using $C = 4$ as the number of clusters, and $m = 1.1$, Fuzzy-SKWIC converged after 27 iterations, resulting in a partition that closesly resembles the distribution of the documents with respect to their true categories.

The class distribution is shown in Table 1.3 and the six most relevant keywords for each cluster are listed in Table 1.4. The highly relevant keywords (Top 2 or 3) are consistent with those obtained using the crisp version. The partition obtained using the fuzzy SKWIC (Table 1.3) is slightly better than the one obtained using the crisp SKWIC (Table 1.1). The partition of the documents of Class 2 still shows the same number of classification errors as in the crisp case. However, a careful examination of the misclassified documents shows that these documents have high membership degrees in more than one cluster, and thus should not be assigned one simple label. Thus, the class distribution in Table 1.3 would greatly improve if the groundtruth labeling was soft from the start. The following excerpt illustrates the soft labels which are automatically computed by Fuzzy SKWIC. They clearly show a documents which is *Mostly about Entertainment, but Somewhat also relating to Business.* Hence in addition to relevant keywords which provide a *short summary* for each cluster, Fuzzy SKWIC can generate a richer soft labeling of the text documents that can aid in retrieval.

**Excerpt from Document 70** [**soft labels:** Business = 85% ($u_{0j}$ = 0.853), Entertainment= 14% ($u_{1j}$ = 0.140), News = 0.5% ($u_{2j}$ = 0.005), Sports = 0.3% ($u_{3j}$ = 0.003)] :

*LOS ANGELES (Reuters) - Ifilm and Pop.com, the would-be Web site backed by film makers Steven Spielberg, Ron Howard and other Hollywood moguls, have ended talks to merge, according to an e-mail sent to Ifilm employees on Friday. ... "The companies will continue to enjoy many overlapping shareholder and personal relationships," the memo said. Industry observers said the founders of Pop.com, which has never aired a single show or launched its Web site, are looking for a graceful exit strategy out of the venture, which has been plagued by infighting and uncertainty about the company's direction and business plan...*

TABLE 1.3. Distribution of the 50 documents from each class into the 4 clusters computed by fuzzy SKWIC

| | **Cluster 1** | **Cluster 2** | **Cluster 3** | **Cluster 4** |
|---|---|---|---|---|
| | **(business)** | **(entertainment)** | **(news)** | **(sports)** |
| class 1 | **48** | 1 | 1 | 0 |
| class 2 | 7 | **31** | 5 | 7 |
| class 3 | 2 | 1 | **47** | 0 |
| class 4 | 0 | 0 | 3 | **47** |

TABLE 1.4. Term relevance for the top six relevant words in each cluster computed by fuzzy SKWIC

| **Cluster # 1** | | **Cluster # 2** | | **Cluster # 3** | | **Cluster # 4** | |
|---|---|---|---|---|---|---|---|
| $v_{1(k)}$ | $w_{(k)}$ | $v_{2(k)}$ | $w_{(k)}$ | $v_{3(k)}$ | $w_{(k)}$ | $v_{4(k)}$ | $w_{(k)}$ |
| 0.029 | compani | 0.031 | film | 0.016 | polic | 0.025 | game |
| 0.016 | percent | 0.012 | star | 0.011 | govern | 0.015 | season |
| 0.011 | share | 0.010 | week | 0.010 | state | 0.010 | plai |
| 0.010 | expect | 0.008 | dai | 0.009 | offici | 0.009 | york |
| 0.008 | market | 0.008 | peopl | 0.009 | nation | 0.009 | open |
| 0.008 | stock | 0.008 | open | 0.009 | sai | 0.009 | run |

### *5.2 Simulation Results on 20 Newsgroups Data*

The second set of experiments is based on the 20 newsgroups data set [Pro93]. This data set is a collection of 20,000 messages, collected from 20 different netnews newsgroups. One thousand messages from each of the twenty newsgroups were chosen at random and partitioned by newsgroup

TABLE 1.5. 20 Class Descriptions

| Class | Class Descriptions | Class | Class Descriptions |
|---:|---|---:|---|
| 1 | alt.atheism | 11 | rec.sport.hockey |
| 2 | comp.graphics | 12 | sci.crypt |
| 3 | comp.os.ms-windows.misc | 13 | sci.electronics |
| 4 | comp.sys.ibm.pc.hardware | 14 | sci.med |
| 5 | comp.sys.mac.hardware | 15 | sci.space |
| 6 | comp.windows.x | 16 | soc.religion.christian |
| 7 | misc.forsale | 17 | talk.politics.guns |
| 8 | rec.autos | 18 | talk.politics.mideast |
| 9 | rec.motorcycles | 19 | talk.politics.misc |
| 10 | rec.sport.baseball | 20 | talk.religion.misc |

name. The list of newsgroups from which the messages were chosen is shown in Table 1.5. The documents were first preprocessed: This included stripping each news message from the e-mail header and special tags, then eliminating stop words and finally stemming words to their root form using the *rainbow* software package [McC96]. Next, words were sorted based on their IDF values. Finally, The number of keywords was reduced by selecting them based on setting a minimum threshold on their sorted IDF values, so as not to exceed a maximum number of words. Since several documents end up with none of the words that were selected, these documents are not considered for clustering. We will first present a discussion of the results obtained on a subset of 2000 documents from the 20 newsgroups data set. This data set is called the *mini newsgroup data set* [McC96]. Then we discuss the results on the entire 20 newsgroups data set.

### Simulation Results on Mini Newsgroups Data using SKWIC

After pre-processing, 449 words were selected based on IDF. Consequently, there were 1730 documents with at least one of these selected keywords. The documents were clustered by SKWIC into $C = 40$ clusters. Note that we arbitrarily chose this number because the actual messages may be categorized better with more clusters. In other words, there is no guarantee that the labeled documents really come from $K = 20$ different categories, since the labelling was done based on the newsgroup name. Moreover, there is no control over messages that may be sent to a particular newsgroup since their topic may differ from the majority in that newsgroup, or even be more similar to a completely different newsgroup.

Table 1.6 shows the class distribution of the 40 clusters discovered by SKWIC. The columns correspond to the class indices which can be mapped to a complete class description using Table 1.5. In general, each row shows one or a few large values, which indicates that the algorithm succeeds in partitioning the majority of same newsgroup documents into a few homogenous clusters according to the specific nature of the documents.

Table 1.7 displays the cluster cardinalities, as well as the top 10 relevant keywords for each cluster, sorted in decreasing order of their relevance weights in each cluster. Note how the relevance weights may vary drastically between different clusters, and this has a significant effect on the weighted distance computations, and hence affect the final partitioning of the documents. By looking at which keywords have highest relevance in a given cluster, and their relevance values, it is possible to roughly deduce the nature of the newsgroup messages that fall into one particular cluster. For example some cluster keyword relevances seem to suggest a stream of discussions that are specific to either a certain event that occured or to a particular issue that grabbed the attention of a subset of participants in a certain newsgroup. Consequently, it can also be seen how some of these clusters can be formed from documents from *distinct* newsgroups because the messages seemed to relate to similar issues that cross different newsgroups. Several such *mixed* clusters can be formed out of documents that cross the boundary between different politics groups, between different religion groups, and even between both politics and religion groups, ..., etc.

Table 1.6 shows some clusters that include documents from different, yet *related* newsgroups. For instance Cluster No. 3 seems to group several documents (61) from all 5 comp. newsgroups ( but with the majority from the *comp.graphics* newsgroup), as well as the sci.electronics (8) and sci.med (6), but suprisingly also some from soc.religion.christian (7) and some from talk.religion.misc (7). Table 1.7 list the top 10 relevant keywords for this cluster which are indicative of the type of content in messages from the comp. and some of the sci. groups, but not necessarily the religion groups. For example, some of the sci.space documents assigned to this cluster speak about solar and lunar images, hence the affinity to graphics. Another message from the talk.religion.misc newsgroup, was assigned to cluster 3 because of some content relating to computers. It had the following quote in the sender's signature: *"A system admin's life is a sorry one. The only advantage he has over Emergency Room doctors is that malpractice suits are rare. On the other hand, ER doctors never have to deal with patients installing new versions of their own innards!"* Here is an excerpt from another message from the talk.religion.misc newsgroup, that was assigned to cluster 3 because of the scientific rethoric (which pulled it towards the comp. and sci. documents in cluster 3): *"This, again, is a belief, not a scientific premise. The original thread referred specifically to "scientific creationism. This means whatever theory or theories you propose must be able to be judged by the scientific method..."*.

There were also several messages concentrating on a major event during that period (Waco's battle), that were assigned to Cluster No. 3, mainly because of the presence of one of the relevant keywords (*semi*). Here is one of the excerpts: " *...in other words faith in a .357 is far stronger than faith in a God providing a miracle for his followers. Interesting. Now, if*

*David Korresh was God, why couldn't he use lightning instead of semi-automatic rifles? ...".* This example illustrates a typical example where the same keyword (*semi*) may have different meanings depending on context.

Just like a cluster can group documents from several related newsgroups, a particular newsgroup may be split into two or more clusters according to the specific topic of the documents. For example, the rec.sport.hockey newsgroup is split over Clusters No. 20 and 21, as can be seen in Table 1.6. Cluster 20 contains more documents from the rec.sport.baseball group, while Cluster No. 21 is more specific of hockey. Table 1.7 reveals completely different keyword distributions and weights for these two clusters, indicating different topics.

Table 1.6 also shows some small clusters with documents from a few newsgroups. For instance, Cluster No. 38 has only 31 documents mostly from the three newsgroups, alt.atheism, soc.religion.christian, talk.religion.misc, and even talk.politics.mideast. It indicates a more specific set of news messages. For example, here is an excerpt from a message from the talk.politics.mideast newsgroups that was assigned in cluster 38) because of the presence of religious words: *".... and judgement it is. Until such time as it recognizes that *any* religiously based government is racist, exclusionary and simply built on a philosophy of "separate but equal" second-class treatment of minorities, it will continue to be known for its bias. If Jewish nationalism is racism, so is Islam; anywhere where people are allotted "different rights" according to race, religion or culture is "racist".*

Some clusters (for instance Cluster No. 0 in Table 1.6) contain documents from almost all newsgroups. Careful examination of some of these documents revealed that most of them do not fit in any of the existing clusters. In fact, their topics are so scattered, that they do not form enough of a consencus to form valid clusters. Hence, they can be considered as *noise* documents that fall into a *noise magnet* cluster that attracts all noise documents that are not strongly typical of any of the other good clusters. These are documents that lie far away or barely on the border of other clusters (see Section 4). In fact Table 1.7 shows that the top 10 relevant keywords have *equally low* relevance weights. In general, the keywords, paired with their relevance weights can be used to infer an automatic (unsupervised) labeling of document clusters.

Finally we note that some documents are grouped together based solely on commonality of their keyword frequencies. The bag of words model is known not to capture the semantics of text. It does not distinguish between different contexts sufficiently to be able to infer that even the same keyword may bear a different meaning. However this model is much less costly than alternative approaches based on Latent Semantic Analysis (LSA) which may be prohibitively costly for huge, dynamic text collections.

**Simulation Results with Fuzzy SKWIC**

Table 1.8 shows the class distribution of the 40 clusters discovered by Fuzzy-SKWIC, with the columns corresponding to the class indices with

TABLE 1.6. SKWIC Results: Distribution of the Mini Newsgroup documents from the 40 clusters into 20 Prelabeled classes

| Cluster | Classes | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 0 | 2 | 2 | 8 | 12 | 8 | 4 | 5 | 14 | 8 | 6 | 4 | 6 | 7 | 8 | 3 | 1 | 1 | 1 | 11 | 9 |
| 1 | 14 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0 | 1 | 0 | 4 |
| 2 | 3 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 3 | 0 | 4 | 1 | 0 | 0 | 3 | 5 | 3 | 4 | 1 |
| 3 | 3 | 26 | 7 | 11 | 11 | 6 | 1 | 0 | 7 | 1 | 2 | 0 | 8 | 6 | 3 | 7 | 2 | 0 | 0 | 7 |
| 4 | 2 | 1 | 10 | 0 | 1 | 5 | 1 | 0 | 0 | 1 | 4 | 4 | 1 | 3 | 2 | 0 | 0 | 2 | 2 | 3 |
| 5 | 1 | 0 | 3 | 1 | 3 | 1 | 1 | 3 | 6 | 0 | 0 | 1 | 1 | 6 | 3 | 4 | 0 | 1 | 1 | 1 |
| 6 | 9 | 2 | 3 | 3 | 3 | 17 | 1 | 2 | 0 | 1 | 0 | 2 | 4 | 0 | 1 | 0 | 1 | 2 | 1 | 8 |
| 7 | 1 | 0 | 5 | 12 | 4 | 1 | 4 | 0 | 1 | 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 3 | 1 | 0 |
| 8 | 1 | 2 | 1 | 1 | 2 | 1 | 8 | 0 | 1 | 0 | 0 | 3 | 8 | 2 | 5 | 0 | 1 | 0 | 0 | 2 |
| 9 | 0 | 1 | 0 | 5 | 9 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 10 | 0 | 6 | 2 | 3 | 4 | 2 | 1 | 2 | 4 | 1 | 0 | 15 | 1 | 1 | 1 | 2 | 3 | 3 | 1 | 0 |
| 11 | 3 | 1 | 1 | 0 | 0 | 2 | 3 | 3 | 3 | 12 | 5 | 1 | 0 | 5 | 2 | 3 | 2 | 4 | 4 | 2 |
| 12 | 0 | 4 | 1 | 1 | 9 | 2 | 5 | 1 | 0 | 0 | 1 | 1 | 8 | 2 | 1 | 0 | 6 | 0 | 0 | 0 |
| 13 | 6 | 2 | 0 | 2 | 1 | 8 | 5 | 1 | 1 | 4 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 14 | 1 | 2 | 2 | 4 | 0 | 0 | 2 | 4 | 2 | 2 | 1 | 2 | 2 | 2 | 3 | 0 | 4 | 6 | 4 | 4 |
| 15 | 0 | 2 | 4 | 1 | 0 | 0 | 1 | 4 | 1 | 0 | 2 | 1 | 0 | 0 | 2 | 21 | 0 | 0 | 0 | 2 |
| 16 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 9 | 2 | 0 | 0 | 2 | 6 | 0 | 4 | 1 | 11 | 0 | 6 | 4 |
| 17 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 4 | 8 | 6 | 1 | 1 | 0 | 3 | 0 | 4 | 3 | 1 | 3 | 2 |
| 18 | 3 | 1 | 1 | 0 | 0 | 0 | 0 | 3 | 2 | 1 | 0 | 1 | 0 | 2 | 1 | 2 | 3 | 4 | 1 | 6 |
| 19 | 1 | 0 | 2 | 0 | 0 | 0 | 3 | 1 | 1 | 4 | 6 | 2 | 2 | 0 | 2 | 1 | 2 | 0 | 1 | 1 |
| 20 | 0 | 2 | 2 | 0 | 1 | 0 | 0 | 2 | 4 | 8 | 11 | 1 | 1 | 1 | 1 | 0 | 1 | 2 | 3 | 1 |
| 21 | 2 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 16 | 1 | 0 | 3 | 0 | 1 | 1 | 1 | 1 | 1 |
| 22 | 1 | 0 | 0 | 2 | 1 | 5 | 4 | 1 | 1 | 1 | 4 | 4 | 6 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 23 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 2 | 2 | 0 | 5 | 3 | 1 | 1 | 1 | 1 | 0 | 16 | 0 |
| 24 | 3 | 5 | 3 | 2 | 3 | 1 | 3 | 1 | 5 | 3 | 1 | 3 | 2 | 0 | 3 | 2 | 2 | 1 | 1 | 3 |
| 25 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 2 | 0 | 0 | 2 | 0 | 25 | 0 | 0 |
| 26 | 0 | 1 | 1 | 2 | 1 | 2 | 0 | 2 | 1 | 4 | 10 | 1 | 0 | 15 | 5 | 0 | 1 | 8 | 2 | 1 |
| 27 | 3 | 0 | 1 | 0 | 3 | 4 | 2 | 0 | 0 | 2 | 0 | 1 | 0 | 13 | 3 | 2 | 3 | 2 | 1 | 3 |
| 28 | 1 | 0 | 3 | 0 | 0 | 0 | 3 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 11 | 1 | 8 | 2 |
| 29 | 2 | 0 | 4 | 0 | 0 | 0 | 2 | 1 | 7 | 0 | 2 | 1 | 0 | 0 | 11 | 0 | 0 | 0 | 1 | 1 |
| 30 | 1 | 3 | 2 | 0 | 4 | 1 | 1 | 0 | 2 | 4 | 1 | 1 | 1 | 3 | 6 | 1 | 6 | 0 | 2 | 1 |
| 31 | 1 | 2 | 1 | 1 | 1 | 3 | 0 | 3 | 2 | 3 | 2 | 5 | 1 | 0 | 2 | 4 | 5 | 3 | 2 | 2 |
| 32 | 3 | 3 | 3 | 1 | 1 | 1 | 0 | 6 | 1 | 9 | 4 | 1 | 0 | 0 | 3 | 0 | 5 | 3 | 1 | 3 |
| 33 | 1 | 5 | 0 | 0 | 0 | 2 | 3 | 2 | 0 | 0 | 1 | 1 | 0 | 1 | 7 | 2 | 3 | 2 | 4 | 5 |
| 34 | 5 | 0 | 3 | 1 | 2 | 2 | 2 | 3 | 2 | 0 | 1 | 7 | 1 | 3 | 2 | 1 | 0 | 1 | 1 | 0 |
| 35 | 1 | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 0 | 2 | 4 | 2 | 2 | 1 | 1 | 7 | 1 | 4 | 2 | 2 |
| 36 | 0 | 3 | 3 | 9 | 3 | 1 | 1 | 0 | 0 | 1 | 4 | 1 | 2 | 0 | 5 | 3 | 0 | 1 | 1 | 0 |
| 37 | 1 | 1 | 0 | 1 | 3 | 3 | 2 | 5 | 9 | 3 | 0 | 4 | 2 | 2 | 3 | 0 | 2 | 1 | 5 | 1 |
| 38 | 7 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 1 | 0 | 1 | 0 | 6 | 1 | 2 | 0 | 10 |
| 39 | 1 | 1 | 2 | 1 | 5 | 0 | 2 | 0 | 2 | 0 | 0 | 1 | 2 | 7 | 2 | 1 | 3 | 1 | 1 | 1 |

complete descriptions listed in Table 1.5. Table 1.9 displays the cluster fuzzy cardinalities($\sum_{j=1}^{N} \mu_{ij}$), as well as the top 10 relevant keywords for each cluster, sorted in decreasing order of their relevance weights in each cluster. Table 1.8 shows a more homogenous class distribution per cluster, indicating a fewer number of documents that risk getting misplaced in a cluster just because they lie on areas of overlap. This is because, fuzzy memberships develop the partition in a softer and more gradual manner, and hence avoid the early commitment of documents to a specific cluster, that occurs with hard 0 or 1 memberships. In fact, it is easier to recognize several meaningful clusters in Table 1.8 with generally larger number of documents from the same newsgroup, and verify that their relevant keywords, in Table 1.9, are more consistent with the newsgroup's nature than corresponding crisp clusters in Table 1.6. For example compare the *sci.medical* cluster (No. 27 in both tables)'s relevant keywords. Other clusters which are easy to delineate include the two atheism clusters (Nos. 1 and 6), the politics.guns cluster (No. 18), the politics.misc cluster(No. 23), the politics.mideast cluster(No. 25), and the religion.christian cluster(No.

TABLE 1.7. SKWIC Results: Cardinality and Term relevance for the top ten relevant words in each cluster

| Cluster | Card | Relevant words |
|---|---|---|
| 0 | 120 | abort(0.1127); ford(0.0889); ec(0.0745); matt(0.0684); desktop(0.0638); coverag(0.0625); gordon(0.0554); backup(0.0476); er(0.0387); hill(0.0340); |
| 1 | 25 | atheism(0.8042); trap(0.0653); wisc(0.0228); smart(0.0157); protest(0.0071); dedic(0.0045); ownership(0.0038); absurd(0.0036); arriv(0.0036); probabl(0.0035); |
| 2 | 34 | senat(0.3564); dozen(0.2000); upset(0.1287); corrupt(0.1211); newspap(0.0164); motor(0.0161); remind(0.0076); loui(0.0067); weird(0.0062); pair(0.0057); |
| 3 | 108 | cpu(0.1799); gif(0.1609); ct(0.0983); intel(0.0678); semi(0.0590); geneva(0.0541); tu(0.0517); adob(0.0421); sharewar(0.0294); ch(0.0277); |
| 4 | 42 | app(0.3982); marc(0.1896); fortun(0.1554); ottawa(0.1136); sequenc(0.0179); invent(0.0096); survei(0.0068); forev(0.0053); ration(0.0051); visibl(0.0050); |
| 5 | 37 | nec(0.3993); babi(0.1853); johnson(0.1279); plate(0.0616); radiat(0.0359); hang(0.0272); panel(0.0216); complaint(0.0151); intens(0.0131); ladi(0.0088); |
| 6 | 60 | motif(0.3176); mathew(0.2737); byte(0.1284); grab(0.0719); satisfi(0.0276); entri(0.0202); minim(0.0134); soldier(0.0113); button(0.0110); dedic(0.0082); |
| 7 | 40 | jumper(0.3723); mm(0.2700); sea(0.1036); label(0.0795); quantum(0.0586); aa(0.0279); interrupt(0.0062); er(0.0049); tube(0.0038); avail(0.0038); |
| 8 | 38 | batteri(0.2963); modul(0.1916); blank(0.1426); filter(0.0753); astronomi(0.0501); intens(0.0425); phase(0.0334); accus(0.0214); tune(0.0166); analog(0.0096); |
| 9 | 23 | simm(0.9034); depth(0.0444); phil(0.0074); slot(0.0043); panel(0.0040); macintosh(0.0030); horizont(0.0030); dale(0.0023); hill(0.0022); sea(0.0020); |
| 10 | 52 | privaci(0.2980); databas(0.1988); slot(0.0938); confer(0.0464); mc(0.0392); ration(0.0329); quiet(0.0265); angl(0.0259); pd(0.0257); caught(0.0165); |
| 11 | 56 | dare(0.1859); strike(0.1221); absurd(0.1095); glad(0.0827); hadn(0.0809); dale(0.0643); staff(0.0597); suck(0.0362); wise(0.0302); favorit(0.0175); |
| 12 | 42 | horizont(0.3209); camera(0.2479); audio(0.2222); tube(0.1018); mess(0.0082); disappear(0.0080); angl(0.0060); speaker(0.0049); filter(0.0045); advertis(0.0039); |
| 13 | 35 | edition(0.2639); height(0.2542); default(0.2137); negoti(0.0582); movi(0.0272); categori(0.0208); trip(0.0161); harri(0.0111); tu(0.0104); disappear(0.0089); |
| 14 | 47 | brown(0.3346); suitabl(0.2316); pay(0.1240); ownership(0.0591); incom(0.0573); ran(0.0286); tune(0.0271); dog(0.0096); ch(0.0086); resid(0.0070); |
| 15 | 41 | bless(0.3348); canon(0.2113); atho(0.1692); liter(0.0832); advertis(0.0390); clh(0.0185); vers(0.0155); exclud(0.0143); ot(0.0103); gospel(0.0065); |
| 16 | 47 | atf(0.3067); detector(0.2603); cop(0.1552); radar(0.1205); laser(0.0462); duti(0.0202); border(0.0069); broke(0.0054); trap(0.0053); tear(0.0039); |
| 17 | 40 | cornell(0.3576); pm(0.2517); philosophi(0.0956); shaft(0.0806); cloth(0.0398); england(0.0235); fee(0.0212); drink(0.0083); crew(0.0073); ident(0.0067); |
| 18 | 31 | counter(0.4446); gospel(0.1541); drink(0.0966); deliber(0.0668); disput(0.0474); stretch(0.0444); excess(0.0073); impact(0.0061); tear(0.0061); bias(0.0061); |
| 19 | 29 | miller(0.4506); detroit(0.2479); diego(0.1402); francisco(0.0317); loui(0.0209); bai(0.0078); walker(0.0073); harri(0.0058); psychologi(0.0057); split(0.0046); |
| 20 | 41 | penalti(0.2904); cap(0.1896); worst(0.1142); prism(0.1005); saturdai(0.0766); impact(0.0586); uh(0.0269); fourth(0.0200); capit(0.0096); circumst(0.0065); |
| 21 | 31 | leaf(0.6507); buffalo(0.1242); battl(0.1056); laugh(0.0164); bright(0.0083); ot(0.0067); sad(0.0064); bai(0.0060); hawk(0.0055); pen(0.0045); |
| 22 | 33 | keyboard(0.5893); pen(0.1421); pgp(0.0754); transform(0.0353); lawyer(0.0246); experienc(0.0213); divid(0.0188); england(0.0107); macintosh(0.0061); clone(0.0049); |
| 23 | 35 | cramer(0.3258); clayton(0.1424); accuraci(0.1346); optilink(0.1327); gai(0.0815); survei(0.0587); male(0.0449); mutual(0.0131); bi(0.0049); craig(0.0044); |
| 24 | 47 | msu(0.3999); parallel(0.1671); premis(0.0667); corner(0.0530); onlin(0.0386); exclus(0.0328); cooper(0.0322); bound(0.0293); pixel(0.0262); floor(0.0216); |
| 25 | 34 | armenian(0.4627); turk(0.1566); armenia(0.1145); turkei(0.0949); villag(0.0473); plane(0.0236); border(0.0207); extermin(0.0115); civilian(0.0089); soldier(0.0085); |
| 26 | 57 | sick(0.3158); diet(0.2141); dick(0.1276); graduat(0.1114); huh(0.0391); roughli(0.0215); muscl(0.0149); harder(0.0112); reserv(0.0087); decent(0.0079); |
| 27 | 43 | symptom(0.2385); clue(0.1545); psychologi(0.1261); deriv(0.0902); magic(0.0572); med(0.0454); sad(0.0331); core(0.0288); hide(0.0286); notion(0.0158); |
| 28 | 34 | assault(0.5312); packet(0.1914); followup(0.1068); influenc(0.0619); emerg(0.0067); sentenc(0.0060); exercis(0.0052); girl(0.0047); evil(0.0047); promot(0.0042); |
| 29 | 32 | digex(0.7346); hawk(0.1200); seat(0.0765); joseph(0.0041); intens(0.0031); gear(0.0029); scratch(0.0027); rear(0.0025); carry(0.0025); motor(0.0023); |
| 30 | 40 | planet(0.3477); editor(0.2376); chemic(0.1800); calcul(0.0523); journal(0.0451); newspap(0.0101); atmospher(0.0086); francisco(0.0058); seat(0.0055); mc(0.0043); |
| 31 | 43 | univ(0.2037); anymor(0.1920); walker(0.1360); centr(0.0991); va(0.0912); ridicul(0.0786); crack(0.0185); numer(0.0180); shaft(0.0085); rotat(0.0080); |
| 32 | 48 | superior(0.1537); craig(0.1310); injuri(0.1250); prison(0.1053); incorrect(0.0796); ideal(0.0607); era(0.0486); silver(0.0433); punish(0.0251); string(0.0239); |
| 33 | 39 | purdu(0.4589); apollo(0.2807); solar(0.0791); attornei(0.0461); broke(0.0350); destruct(0.0093); lawyer(0.0058); pair(0.0047); probabl(0.0037); declar(0.0036); |
| 34 | 37 | dont(0.5735); session(0.2218); attract(0.0490); billion(0.0470); worship(0.0219); prism(0.0057); sharewar(0.0050); desktop(0.0037); med(0.0037); implement(0.0032); |
| 35 | 39 | pp(0.3959); credit(0.2320); relationship(0.1024); implement(0.0358); shown(0.0356); clh(0.0235); vers(0.0184); advis(0.0151); graduat(0.0131); declar(0.0074); |
| 36 | 38 | gatewai(0.4673); mon(0.2335); jpl(0.0961); bi(0.0624); phil(0.0371); interrupt(0.0102); buck(0.0060); experienc(0.0045); pitt(0.0042); utexa(0.0038); |
| 37 | 48 | oil(0.1880); alot(0.1690); bag(0.1136); blind(0.1120); weird(0.0957); pair(0.0420); environment(0.0277); eh(0.0272); engag(0.0213); neat(0.0203); |
| 38 | 31 | dwyer(0.2897); judgem(0.2469); horu(0.1252); mchp(0.1252); sni(0.1252); greatest(0.0269); infinit(0.0042); punish(0.0032); walker(0.0032); crack(0.0023); |
| 39 | 33 | iastat(0.4820); tast(0.2779); instrum(0.0955); sector(0.0446); intel(0.0067); filter(0.0061); cloth(0.0028); shield(0.0027); sequenc(0.0026); profit(0.0026); |

35).

Soft memberships allow a document to belong to several clusters simultaneously, and hence provide a richer model in the areas of overlap. We will not show examples in this section, since we have already illustrated how Fuzzy-SKWIC succeeds in providing richer soft labeling for the Web documents in Section 5.1. What is worth mentionning in the fuzzy case, is that as a result of assigning soft membership degrees to the documemts in each cluster, the noise documents which are roughly equally far from the majority of good clusters, get assigned similar soft memberships in all clusters. Hence they are discouraged from *conspiring* against one of the clusters as in the crisp partitioning framework, where they can acquire a *whole* membership of 1 in a given cluster because of arbitrary crisp assignment based on minimum (within $\epsilon$) distance. This means that, generally, noise documents will have almost equal memberships ($\frac{1}{C}$ in all clusters, hence their influence on good clusters is broken up into smaller equal pieces instead of a whole sum. Consequently, their net effect on the resulting partition and all estimated parameters (since everything is weighted by the memberships) gets diluted, and this is what makes our *soft* partitioning strategy more *robust* to noise. A direct consequence of this fact, is that there is no longer a big noise cluster grouping several documents from all newsgroups as in the crisp case (Cluster No. 3).

We note that despite the *softness* of the memberships, the clusters which are very homogenous in the nature of their documents, end up with almost *crisp* 0-1 memberships. Hence the crisp partition is a special case of soft partitioning that does emerge when there is no strong overlap between different clusters.

We have further performed clustering using unweighted keyword based techniques: K Means and the Fuzzy C Means, (both with cosine based distance) and have noticed that both crisp and fuzzy SKWIC tend to outperform their unweighted counterparts. For instance, the noise cluster that grabs documents from all different newsgroups gets even larger. To summarize, K Means lies on the least favorable side of the spectrum because it has no way of adapting different clusters to capture different relevance degrees in their keywords, nor different membership degrees of their documents. SKWIC is able to model different keyword relevance degrees depending on the cluster, but cannot model gradual degrees of membership of documents. The Fuzzy C Means fails to model different cluster-dependent keyword relevance degrees but can model gradual degrees of membership of documents. Hence both SKWIC and Fuzzy C Means have complementary but exclusive strengths that make them provide richer partition models. However, Fuzzy SKWIC lies on the most favorable side of the spectrum because it is able to provide both dynamic soft degrees in the keyword relevance values and in the cluster memberships, and can be thus considered to perform simultaneous partitioning in two different hyperspaces: the document space to capture *spatial* document organization, and the keyword space to capture

TABLE 1.8. Fuzzy-SKWIC Results: Distribution of the Mini Newsgroup documents from 40 clusters into 20 Prelabeled classes

| Cluster | Classes | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** | **20** |
| **0** | 4 | 3 | 2 | 3 | 2 | 23 | 5 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| **1** | 14 | 0 | 0 | 0 | 0 | 3 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 4 |
| **2** | 3 | 2 | 5 | 8 | 2 | 0 | 0 | 0 | 5 | 3 | 2 | 3 | 1 | 1 | 2 | 3 | 1 | 1 | 1 | 1 |
| **3** | 0 | 18 | 3 | 2 | 5 | 1 | 1 | 0 | 4 | 0 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| **4** | 1 | 1 | 9 | 1 | 1 | 4 | 1 | 2 | 2 | 4 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| **5** | 4 | 0 | 2 | 1 | 2 | 1 | 3 | 2 | 7 | 0 | 0 | 1 | 4 | 1 | 0 | 1 | 1 | 2 | 1 | 1 |
| **6** | 10 | 2 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 6 |
| **7** | 1 | 1 | 4 | 13 | 7 | 0 | 1 | 1 | 0 | 2 | 0 | 3 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| **8** | 6 | 2 | 7 | 1 | 3 | 0 | 0 | 1 | 3 | 2 | 0 | 14 | 3 | 4 | 4 | 2 | 4 | 0 | 0 | 5 |
| **9** | 0 | 0 | 0 | 5 | 10 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **10** | 0 | 5 | 0 | 0 | 0 | 2 | 2 | 0 | 2 | 2 | 2 | 9 | 0 | 1 | 0 | 0 | 2 | 3 | 2 | 1 |
| **11** | 2 | 0 | 1 | 7 | 0 | 3 | 2 | 2 | 2 | 4 | 4 | 0 | 4 | 9 | 1 | 1 | 3 | 3 | 3 | 6 |
| **12** | 0 | 4 | 1 | 1 | 6 | 2 | 9 | 1 | 1 | 0 | 1 | 1 | 9 | 3 | 1 | 0 | 3 | 1 | 1 | 0 |
| **13** | 3 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 7 | 0 | 5 | 4 | 1 | 1 | 0 | 1 | 0 | 0 | 2 |
| **14** | 2 | 2 | 2 | 6 | 2 | 2 | 5 | 17 | 6 | 4 | 3 | 0 | 3 | 1 | 8 | 0 | 1 | 2 | 3 | 1 |
| **15** | 2 | 2 | 3 | 3 | 3 | 0 | 1 | 4 | 3 | 0 | 3 | 2 | 0 | 0 | 1 | 16 | 2 | 0 | 0 | 4 |
| **16** | 0 | 1 | 0 | 0 | 2 | 0 | 2 | 7 | 3 | 0 | 0 | 0 | 7 | 0 | 6 | 0 | 3 | 0 | 2 | 0 |
| **17** | 0 | 0 | 0 | 3 | 1 | 2 | 1 | 5 | 9 | 7 | 1 | 0 | 0 | 4 | 1 | 3 | 2 | 1 | 4 | 2 |
| **18** | 1 | 1 | 3 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 2 | 0 | 1 | 0 | 1 | 21 | 1 | 8 | 7 |
| **19** | 1 | 2 | 2 | 0 | 1 | 3 | 1 | 2 | 2 | 7 | 19 | 1 | 1 | 3 | 4 | 1 | 4 | 4 | 5 | 0 |
| **20** | 0 | 1 | 3 | 2 | 6 | 0 | 3 | 2 | 1 | 4 | 8 | 1 | 4 | 0 | 1 | 0 | 0 | 0 | 2 | 1 |
| **21** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 14 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 |
| **22** | 0 | 0 | 3 | 7 | 10 | 6 | 6 | 0 | 1 | 3 | 0 | 3 | 8 | 1 | 0 | 0 | 1 | 0 | 0 | 2 |
| **23** | 1 | 0 | 2 | 0 | 0 | 0 | 0 | 2 | 0 | 2 | 3 | 1 | 1 | 0 | 3 | 0 | 1 | 0 | 17 | 0 |
| **24** | 1 | 0 | 2 | 1 | 4 | 1 | 2 | 5 | 6 | 2 | 2 | 1 | 2 | 1 | 3 | 2 | 3 | 1 | 2 | 2 |
| **25** | 2 | 4 | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 1 | 0 | 4 | 0 | 1 | 1 | 0 | 30 | 0 | 0 |
| **26** | 3 | 1 | 1 | 1 | 0 | 1 | 0 | 2 | 0 | 5 | 6 | 1 | 1 | 6 | 3 | 1 | 3 | 6 | 3 | 3 |
| **27** | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 2 | 4 | 0 | 0 | 34 | 1 | 0 | 1 | 0 | 0 | 0 |
| **28** | 0 | 1 | 0 | 4 | 0 | 3 | 2 | 4 | 1 | 4 | 1 | 5 | 0 | 1 | 4 | 5 | 5 | 2 | 10 | 5 |
| **29** | 1 | 0 | 4 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 11 | 0 | 0 | 0 | 1 | 1 |
| **30** | 1 | 3 | 1 | 0 | 0 | 1 | 4 | 5 | 9 | 3 | 6 | 1 | 2 | 1 | 4 | 1 | 3 | 1 | 1 | 0 |
| **31** | 3 | 2 | 0 | 1 | 1 | 3 | 0 | 2 | 5 | 4 | 2 | 3 | 1 | 4 | 3 | 5 | 6 | 1 | 5 | 3 |
| **32** | 4 | 2 | 2 | 1 | 0 | 1 | 2 | 2 | 2 | 5 | 1 | 4 | 1 | 0 | 4 | 5 | 6 | 6 | 5 | 7 |
| **33** | 1 | 3 | 0 | 0 | 0 | 2 | 3 | 2 | 0 | 0 | 1 | 2 | 0 | 0 | 4 | 2 | 3 | 5 | 5 | 6 |
| **34** | 4 | 2 | 6 | 2 | 1 | 4 | 0 | 2 | 4 | 1 | 1 | 13 | 0 | 4 | 1 | 2 | 1 | 2 | 2 | 1 |
| **35** | 2 | 1 | 2 | 1 | 1 | 2 | 2 | 1 | 2 | 1 | 2 | 2 | 1 | 2 | 0 | 25 | 1 | 8 | 0 | 3 |
| **36** | 2 | 5 | 3 | 1 | 0 | 1 | 4 | 1 | 0 | 1 | 0 | 1 | 3 | 0 | 13 | 1 | 2 | 0 | 1 | 1 |
| **37** | 1 | 8 | 3 | 0 | 6 | 1 | 4 | 5 | 4 | 3 | 1 | 4 | 2 | 1 | 3 | 0 | 4 | 5 | 5 | 4 |
| **38** | 7 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 1 | 0 | 1 | 0 | 5 | 1 | 1 | 0 | 10 |
| **39** | 1 | 1 | 4 | 2 | 9 | 0 | 2 | 1 | 2 | 0 | 0 | 1 | 5 | 1 | 1 | 0 | 2 | 1 | 1 | 3 |

*context.* The context can be inferred for each cluster because it is described in terms of several relevant keywords, and these keywords are further given importance degrees that vary with each cluster. The context stems mainly from the well known fact that it is easier to infer context from *several* keywords simultaneously, than from any single one of the keywords. The relevance weights are expected to further enrich the context description.

### Simulation Results on the 16 Mini Newsgroup Data and Entire 20 Newsgroups Data

With the Mini Newsgroup data set in the previous section, we have noticed that there were several misclassified (or inconsistently assigned) documents that come from the four miscellaneous classes (Nos. 3, 7, 19, 20). Most of these documents have been assumed to have the same groundtruth label (newsgroup name), but their contents do widely vary in topic, in a way that would make some of them more appropriately labeled with other newsgroup names. Therefore, we repeated all experiments after discarding documents from these 4 classes. This means that we removed most of the

TABLE 1.9. Fuzzy-SKWIC Results: Fuzzy Cardinality and Term relevance for the top ten relevant words in each cluster

| Cluster | Card | Relevant words |
|---|---|---|
| 0 | 43.87 | motif(0.5799); default(0.1127); string(0.0882); height(0.0460); depth(0.0448); hang(0.0234); focu(0.0103); button(0.0081); databas(0.0053); byte(0.0052); |
| 1 | 25.52 | atheism(0.9049); wisc(0.0078); trap(0.0049); ownership(0.0043); arriv(0.0042); absurd(0.0041); probabl(0.0040); dedic(0.0040); suitabl(0.0039); declar(0.0036); |
| 2 | 41.52 | gatewai(0.5335); upset(0.1644); mirror(0.0621); bi(0.0606); corrupt(0.0484); phil(0.0301); suck(0.0048); utexa(0.0048); batteri(0.0045); suddenli(0.0044); |
| 3 | 42.36 | gif(0.4690); pixel(0.1592); slot(0.1216); clip(0.1017); pd(0.0606); blank(0.0084); implement(0.0071); domain(0.0058); plane(0.0047); sharewar(0.0045); |
| 4 | 33.55 | app(0.5580); lee(0.2951); decent(0.0500); favorit(0.0040); staff(0.0040); intens(0.0038); tech(0.0037); superior(0.0031); sea(0.0027); bag(0.0026); |
| 5 | 30.58 | nec(0.5792); babi(0.2441); lawyer(0.0545); ladi(0.0111); dog(0.0093); armi(0.0075); odd(0.0064); punish(0.0050); turk(0.0046); joseph(0.0045); |
| 6 | 29.40 | mathew(0.7106); ideal(0.1854); quantum(0.0099); resembl(0.0067); turkei(0.0051); civilian(0.0051); dwyer(0.0041); decent(0.0039); interrupt(0.0037); greatest(0.0036); |
| 7 | 33.67 | jumper(0.5789); quantum(0.2108); interrupt(0.0914); advoc(0.0113); corrupt(0.0079); movi(0.0065); avail(0.0056); label(0.0055); speech(0.0050); convent(0.0050); |
| 8 | 53.64 | privaci(0.2250); fortun(0.1716); carl(0.1527); premis(0.1092); modul(0.0939); perman(0.0411); exclud(0.0297); swap(0.0256); blank(0.0239); habit(0.0060); |
| 9 | 26.37 | simm(0.9450); phil(0.0077); panel(0.0046); slot(0.0045); depth(0.0037); macintosh(0.0031); horizont(0.0031); dale(0.0027); sea(0.0023); wise(0.0017); |
| 10 | 36.54 | databas(0.3113); senat(0.2615); confer(0.2356); caught(0.0332); foreign(0.0284); pa(0.0105); punish(0.0049); fourth(0.0047); forev(0.0046); crack(0.0045); |
| 11 | 61.54 | ct(0.1600); dare(0.1350); pitt(0.1226); gordon(0.1193); absurd(0.0902); ridicul(0.0874); strike(0.0711); hadn(0.0486); newspap(0.0233); clinic(0.0073); |
| 12 | 50.37 | mm(0.2406); audio(0.2052); horizont(0.2031); tube(0.1724); camera(0.0807); disappear(0.0107); mess(0.0072); wa(0.0054); speaker(0.0048); advertis(0.0038); |
| 13 | 39.38 | ec(0.5476); edition(0.1696); diego(0.1348); categori(0.0509); entri(0.0059); disappear(0.0044); pa(0.0044); capit(0.0041); invent(0.0036); greatest(0.0033); |
| 14 | 73.57 | ford(0.2593); brown(0.1063); allen(0.0892); tune(0.0812); dick(0.0778); batteri(0.0716); pay(0.0598); resid(0.0496); plate(0.0360); gear(0.0327); |
| 15 | 49.40 | canon(0.2992); bless(0.2541); er(0.1303); atho(0.1006); liter(0.0632); buffalo(0.0549); clh(0.0127); ot(0.0117); filter(0.0039); sad(0.0037); |
| 16 | 32.48 | detector(0.4793); radar(0.2463); cop(0.1577); duti(0.0265); border(0.0128); laser(0.0124); worker(0.0072); angl(0.0063); trap(0.0051); max(0.0047); |
| 17 | 42.61 | cornell(0.3754); pm(0.2749); drink(0.1067); shaft(0.1004); loui(0.0096); wound(0.0064); austin(0.0055); fee(0.0054); utexa(0.0054); attend(0.0050); |
| 18 | 54.42 | atf(0.3720); assault(0.2372); packet(0.1930); stretch(0.0427); pointer(0.0405); camera(0.0071); threat(0.0059); broke(0.0059); lawyer(0.0058); attornei(0.0047); |
| 19 | 57.56 | pp(0.2393); penalti(0.2060); chemic(0.1558); impact(0.0866); detroit(0.0789); prism(0.0737); worst(0.0538); capit(0.0092); loui(0.0061); circl(0.0056); |
| 20 | 41.42 | backup(0.5167); cap(0.2153); pen(0.0721); laser(0.0657); analog(0.0338); fourth(0.0051); classic(0.0043); devil(0.0038); attend(0.0035); buffalo(0.0029); |
| 21 | 24.31 | leaf(0.8989); bai(0.0083); hawk(0.0076); pen(0.0062); detroit(0.0060); devil(0.0057); leg(0.0055); bright(0.0051); lee(0.0051); carl(0.0039); |
| 22 | 51.57 | cpu(0.4203); keyboard(0.3074); catch(0.0928); macintosh(0.0710); transform(0.0286); clone(0.0097); suitabl(0.0055); speaker(0.0055); blind(0.0053); tech(0.0045); |
| 23 | 34.45 | cramer(0.3586); clayton(0.1669); survei(0.1506); optilink(0.1460); gai(0.0765); mutual(0.0277); male(0.0142); bi(0.0054); ottawa(0.0052); sea(0.0042); |
| 24 | 44.52 | msu(0.4406); chain(0.2240); devil(0.0981); visibl(0.0609); exclus(0.0564); trap(0.0106); negoti(0.0058); oil(0.0043); effici(0.0043); session(0.0042); |
| 25 | 51.48 | armenian(0.3693); tu(0.1333); soldier(0.0877); armenia(0.0873); turk(0.0715); turkei(0.0559); villag(0.0546); plane(0.0363); border(0.0338); extermin(0.0096); |
| 26 | 41.72 | sick(0.5233); counter(0.1695); huh(0.0986); roughli(0.0467); disput(0.0449); bias(0.0113); accus(0.0084); amateur(0.0063); harder(0.0052); walker(0.0047); |
| 27 | 49.52 | diet(0.3122); symptom(0.2068); tast(0.1057); med(0.0869); literatur(0.0442); muscl(0.0411); healthi(0.0392); clinic(0.0134); root(0.0123); mouth(0.0116); |
| 28 | 57.65 | abort(0.3024); matt(0.2011); coverag(0.1483); workstat(0.1115); followup(0.0667); denni(0.0476); protest(0.0078); acknowledg(0.0064); emerg(0.0063); convict(0.0056); |
| 29 | 26.36 | digex(0.9382); intens(0.0041); gear(0.0037); scratch(0.0036); carry(0.0033); motor(0.0032); bound(0.0027); restor(0.0026); carl(0.0025); confer(0.0023); |
| 30 | 44.66 | craig(0.2987); seat(0.2141); editor(0.1603); hawk(0.1180); francisco(0.0753); floor(0.0089); batteri(0.0088); mid(0.0083); rear(0.0067); candid(0.0044); |
| 31 | 47.69 | adob(0.2839); univ(0.2235); anymor(0.1875); johnson(0.0946); laugh(0.0386); va(0.0242); evil(0.0107); exercis(0.0079); weird(0.0059); pa(0.0059); |
| 32 | 40.30 | battl(0.3249); superior(0.2138); hide(0.1514); fee(0.0549); tear(0.0549); dy(0.0221); root(0.0085); deliber(0.0080); forev(0.0078); glad(0.0070); |
| 33 | 42.43 | purdu(0.4496); apollo(0.3028); attornei(0.0690); prison(0.0597); broke(0.0208); destruct(0.0076); lawyer(0.0073); pair(0.0053); probabl(0.0041); declar(0.0039); |
| 34 | 51.53 | dont(0.4483); session(0.1581); pgp(0.1095); england(0.0695); implement(0.0609); worship(0.0521); prism(0.0050); entri(0.0049); desktop(0.0040); australia(0.0039); |
| 35 | 51.93 | geneva(0.3260); graduat(0.1728); male(0.0931); relationship(0.0915); vers(0.0658); clh(0.0561); harri(0.0260); credit(0.0148); adult(0.0106); proven(0.0104); |
| 36 | 38.55 | miller(0.3295); sharewar(0.2967); jpl(0.1575); solar(0.1126); fee(0.0075); marc(0.0052); convent(0.0052); sea(0.0051); psychologi(0.0050); attract(0.0042); |
| 37 | 62.63 | ch(0.2326); su(0.1411); domain(0.0870); planet(0.0779); negoti(0.0597); ownership(0.0579); advertis(0.0549); environment(0.0459); weird(0.0415); marc(0.0403); |
| 38 | 33.43 | dwyer(0.3045); judgem(0.2431); horu(0.1317); mchp(0.1317); sni(0.1317); infinit(0.0045); greatest(0.0039); walker(0.0038); punish(0.0036); crack(0.0025); |
| 39 | 35.46 | iastat(0.4500); intel(0.4329); sector(0.0417); instrum(0.0055); battl(0.0048); forev(0.0043); tast(0.0043); optic(0.0027); shield(0.0026); button(0.0026); |

*difficult* cases so to speak. These include, i) documents that lie in areas of overlap or fuzziness between distinct categories, or (ii) documents that are simply outliers, and hence affect the purity of the resulting partition. After discarding the .misc classes, we noticed similar results in terms of the nature of the clusters, and the richer information provided by the cluster dependent keyword relevance weights, and soft partition. We also noticed a remarkable improvement in the purity of the partition with regard to cluster homogeneity, as compared to including the miscellaneous class documents.

One way to objectively assess the performance of a clustering algorithm when the class labels for $K$ classes are actually known, is based on the average entropy measure of all $C$ clusters, which is defined as follows

$$E = \sum_{i=1}^{C} \frac{N_i}{N} E_i,$$

where

$$E_i = \frac{1}{\log K} \sum_{k=1}^{K} \frac{N_i^k}{N_i} \log \frac{N_i^k}{N_i},$$

is the entropy of the $i^{th}$ cluster, $N_i$ is the size of the $i^{th}$ cluster, and $N_i^k$ is the number of documents from the $k^{th}$ class which are assigned to the $i^{th}$ cluster. Table 1.10 lists the entropies of the partitioning strategies used for the Mini and 20 Newsgroup data, depending on whether the 4 miscellaneous classes are removed.

With all the empirical results and theoretically based conclusions about the data sets used in this chapter in mind, the most important fact to remember is that in our *nonideal* world, *real unlabelled* text data tends to be of the *challenging type* discussed above. This in turn calls for sophisticated techniques that can handle these challenges.

We also note that with the 20 Newsgroups data set, as with almost any manually labeled benchmark document data set, errors in labeling abound (due to subjectivity in labeling, circumstantial reasons, or even noise documents that still end up with an invalid label). Also, documents that cross boundaries between different categories are very common, and always end up with an inadequate label. Hence it is exteremely difficult to judge the quality of an unsupervised clustering technique based on any kind of classification accuracy or *entropy* measure. In fact, our experiments have showed that automatic labeling is often superior to manual labeling, except when identical keywords with different meanings are present. This is where keyword based clustering breaks down because it cannot deal with the semantics of content. For such cases, context can improve clustering results considerably, and this can be handled using Latent Semantic Indexing [DDF$^+$90, BDJ99] for example.

TABLE 1.10. Average Entropies for different categorization strategies of the Newsgroup Data

| | **Mini Newsgroups** | **Mini Newsgroups 16 Class** | **20 Newsgroups** |
|---|---|---|---|
| K Means | 0.797 | 0.771 | 0.865 |
| SKWIC | 0.790 | 0.750 | 0.866 |
| Fuzzy C Means | 0.766 | 0.751 | 0.907 |
| Fuzzy SKWIC | 0.757 | 0.740 | 0.868 |

## 6 Conclusion

In this chapter, we presented a new approach that performs clustering and attribute weighting simultaneously 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 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 categorizing 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.

We have also presented a soft partitioning approach (Fuzzy SKWIC) to handle the inherent fuzziness in text documents, by automatically generating fuzzy or soft labels instead of single-label categorization. This means that a text document can belong to *several* categories with different degrees. The soft approach succeeds in describing documents at the intersection between several categories.

By virtue of the dynamic keyword weighting, and its continuous interaction with distance and membership computations, the proposed approach is able to handle noise documents elegantly by automatically designating one or two *noise magnet* clusters that grab most outliers away from the other clusters.

Compared to variants such as K Means and the Fuzzy C Means, Fuzzy SKWIC is able to provide both dynamic soft degrees in the keyword relevance values and in the cluster memberships, and can be thus considered to perform simultaneous partitioning in two different hyperspaces: the document space to capture *spatial* document organization, and the keyword space to capture *context*. The context stems mainly from the well known fact that it is easier to infer context from a consencus of *several* keywords simultaneously, than from any single one of the keywords. The relevance weights are expected to further enrich the context description.

Our results have also confirmed that Fuzzy SKWIC, is most appropriate to use in cases where the document collection is *challenging*, meaning that it may be limited in terms of the number of documents, and the number of keywords used to infer the labels, and that it may include many *noise* documents and *mixed-topic* documents that blur the boundaries between clusters. Our *nonideal* world abounds with *unlabelled* text data that tends to be of the *challenging type*. Fuzzy SKWIC is one of the unsupervised classification techniques that can handle these challenges.

We also note that with the 20 Newsgroups data set, as with almost any manually labeled benchmark document data set, errors in labeling (due to noise documents, subjectivity, and mixed topic documents that cross boundaries between different categories) are very common. Hence it is exteremely difficult to judge the quality of an unsupervised clustering technique based on any kind of classification accuracy or *entropy* measure. In fact, our experiments have showed that automatic labeling is often superior to manual labeling, except when identical keywords with different meanings are present. This is where keyword based clustering breaks down because it cannot deal with the semantics of content. For such cases, context can improve clustering results considerably, and this may be handled using Latent Semantic Indexing [DDF$^+$90, BDJ99].

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 [KK93, NK96, FK99, NK97], to determine the number of clusters [FK97], to cluster very large data sets [BFR98, FLE00], and to improve initialization [BF98, HOB99, NK00]. Future research includes investigating more scalable extensions that are not sensitive to initialization, and that can determine the optimal number of clusters. We also plan to explore context-dependent information retrieval based on a combination of Fuzzy SKWIC with concepts from fuzzy logic, particularly its ability to *compute with words*.

## 7 Acknowledgments

We thank our student Mrudula Pavuluri for her help with preprocessing some of the text data. Partial support of this work by National Science Foundation CAREER Award IIS 0133415 to Hichem Frigui, and by National Science Foundation CAREER Award IIS 0133948 to Olfa Nasraoui is gratefully acknowledged.

## 8 References

[AD91] H. Almuallim and T. G. Dietterich. Learning with many irrelevant features. In *Ninth National Conference on artificial intelligence*, pages 547–552, 1991.

[BDJ99] Michael W. Berry, Zlatko Drmac, and Elizabeth R. Jessup. Matrices, vector spaces, and information retrieval. *SIAM Review*, 41(2):335–362, 1999.

[Bez81] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York, 1981.

[BF98] Paul S. Bradley and Usama M. Fayyad. Refining initial points for K-Means clustering. In *Proc. 15th International Conf. on Machine Learning*, pages 91–99. Morgan Kaufmann, San Francisco, CA, 1998.

[BFR98] P. S. Bradley, Usama M. Fayyad, and Cory Reina. Scaling clustering algorithms to large databases. In *Knowledge Discovery and Data Mining*, pages 9–15, 1998.

[BL85] C. Buckley and A.F. Lewit. Optimizations of inverted vector searches. In *SIGIR '85*, pages 97–110, 1985.

[CKPT92] 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*, pages 318–329, 1992.

[DDF$^+$90] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[FK97] H. Frigui and R. Krishnapuram. Clustering by competitive agglomeration. *Pattern Recognition*, 30(7):1223–1232, 1997.

[FK99] H. Frigui and R. Krishnapuram. A robust competitive clustering algorithm with applications in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):450–465, May. 1999.

[FLE00] Fredrik Farnstrom, James Lewis, and Charles Elkan. Scalability for clustering algorithms revisited. *SIGKDD Explorations*, 2(1):51–57, 2000.

[FN00] H. Frigui and O. Nasraoui. Simultaneous clustering and attribute discrimination. In *IEEE Conference on Fuzzy Systems*, pages 158–163, San Antonio, Texas, 2000.

[GK79] E. E. Gustafson and W. C. Kessel. Fuzzy clustering with a fuzzy covariance matrix. In *IEEE CDC*, pages 761–766, San Diego, California, 1979.

[HOB99] L. O. Hall, I. O. Ozyurt, and J. C. Bezdek. Clustering with a genetically optimized approach. *IEEE Trans. Evolutionary Computations*, 3(2):103–112, July 1999.

[Hub81] P. J. Huber. *Robust Statistics.* John Wiley & Sons, New York, 1981.

[JKP94] G. John, R. Kohavi, and K. Pfleger. Irrelevant features and the subset selection problem. In *Eleventh International Machine Learning Conference*, pages 121–129, 1994.

[KK93] R. Krishnapuram and J. M. Keller. A possibilistic approach to clustering. *IEEE Trans. Fuzzy Syst.*, 1(2):98–110, May 1993.

[Kor97] R. R. Korfhage. *Information Storage and Retrieval.* Wiley, 1997.

[Kow97] G. Kowalski. *Information retrieval systems-theory and implementations.* Kluwer Academic Publishers, 1997.

[KR92] K. Kira and L. A. Rendell. The feature selection problem: Traditional methods and a new algorithm. In *Tenth National Conference on artificial intelligence*, pages 129–134, 1992.

[KS95] 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*, pages 192–197, 1995.

[McC96] Andrew Kachites McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow, 1996.

[Mla99] D. Mladenic. Text learning and related intelligent agents. *IEEE Expert*, Jul. 1999.

[NK96] O. Nasraoui and R. Krishnapuram. An improved possibilistic c-means algorithm with finite rejection and robust scale estimation. In *North American Fuzzy Information Processing Society Conference*, Berkeley, California, June. 1996.

[NK97] O. Nasraoui and R. Krishnapuram. Clustering using a genetic fuzzy least median of squares algorithm. In *North American Fuzzy Information Processing Society Conference*, Syracuse NY, Sep. 1997.

[NK00] O. Nasraoui and R. Krishnapuram. A novel approach to unsupervised robust clustering using genetic niching. In *IEEE International Conference on Fuzzy Systems*, pages 170–175, New Orleans, 2000.

[Pro93] CMU's WebKB Project. 20 newsgroup data set. http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/news20.html, 1993.

[RK92] L. A. Rendell and K. Kira. A practical approach to feature selection. In *International Conference on machine learning*, pages 249–256, 1992.

[RL87] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.

[Ska94] D. Skalak. Prototype and feature selection by sampling and random mutation hill climbing algorithms. In *Eleventh International Machine Learning Conference (ICML-94)*, pages 293–301, 1994.

[Van89] C.J. VanRijsbergen. *Information Retrieval*. Buttersworth, London, 1989.

[ZEMK97] O. Zamir, O. Etzioni, O. Madani, and R.M. Karp. Fast and intuitive clustering of web documents. In *KDD'97*, pages 287–290, 1997.