Flexible document organization: comparing fuzzy and possibilistic approaches

Flexible Document Organization: Comparing Fuzzy and Possibilistic Approaches

Tatiane M. Nogueira
Department of Computer Science
Federal University of Bahia– Brazil
{tatianenogueira@dcc.ufba.br}

Solange O. Rezende
Institute of Mathematics and Computer Science
University of São Paulo – Brazil
{solange@icmc.usp.br}

Heloisa A. Camargo
Department of Computer Science
Federal University of São Carlos – Brazil
{heloisa@dc.ufscar.br}

***Abstract*—System flexibility means the ability of a system to manage imprecise and/or uncertain information. A lot of commercially available Information Retrieval Systems (IRS) address this issue at the level of query formulation. Another way to make the flexibility of an IRS possible is by means of the flexible organization of documents. Such organization can be carried out using clustering algorithms by which documents can be automatically organized in multiple clusters simultaneously. Fuzzy and possibilistic clustering algorithms are examples of methods by which documents can belong to more than one cluster simultaneously with different membership degrees. The interpretation of these membership degrees can be used to quantify the compatibility of a document with a particular topic. The topics are represented by clusters and the clusters are identified by one or more descriptors extracted by a proposed method. We aim to investigate if the performance of each clustering algorithm can affect the extraction of meaningful overlapping cluster descriptors. Experiments were carried using well-known collections of documents and the predictive power of the descriptors extracted from both fuzzy and possibilistic document clustering was evaluated. The results prove that descriptors extracted after both fuzzy and possibilistic clustering are effective and can improve the flexible organization of documents.**

Keywords*—*fuzzy clustering, possibilistic clustering, flexible organization, documents, information retrieval

I. Introduction

The Information Retrieval (IR), according to [1], aims the development of computer systems for the storage and retrieval of textual information in the form of documents. The main activity of an Information Retrieval System (IRS) is to gather pertinent stored documents that better satisfy the user's information requirements requested by means of a query. Both documents and user's queries must be formally represented in a consistent way and, with this, the IRS can satisfactorily develop the retrieval activity.

One of the main limitations of IRS is that its flexibility has been handled only at the level of query formulation, whereas the document organization is rigidly interpreted by the retrieval mechanism [2]. According to [3], system flexibility means the ability of a system to manage imprecise and/or uncertain information.

To illustrate the usefulness of such a flexibility in the document organization level, consider a context in which news are organized in categories according to their main topic. Consider a news (textual document) with the title "*Experts affirm the adventure sport strengthens heart health*", which addresses complementary topics: *Sports* and *Health*. This news can be assigned to distinct categories: the categories related to the *Sports* topic or the categories related to the *Health* topic. Nevertheless, the cited news deals with both topics simultaneously, which suggests that the assignment of this news to categories that represent both topics would be more appropriate than choosing categories that represents just one of them.

Therefore, supposing that a user is requiring documents of the *Sports* topic, if the cited document is assigned only to the *Health* topic, this document would not be recovered for the user, despite being useful for his/her requirements.

One way to develop the retrieval activity of an IRS is to organize documents in categories by means of clustering. Clustering algorithms group documents that share many terms, what is an indication that the content of these documents is similar. Document clustering is used in a variety of IR applications because if there is a document in a cluster that is relevant to a user, then it is likely that other documents from the same cluster are also relevant [4].

Furthermore, to overcome the drawback concerning multi-topic documents, there are clustering algorithms designed to produce overlapping clustering solutions [5], [6], [7], [8], [9], [10], [11].

The Fuzzy C-Means (FCM) [12] and Possibilistic C-Means (PCM) [13] clustering algorithms are examples of methods by which documents are automatically organized in multiple clusters simultaneously [14], [15], [16], [17], [19], [20]. FCM and PCM clustering algorithms scatter a document collection so that each document may belong to different clusters with different membership degrees. The interpretation of these membership degrees can be used to quantify the compatibility of a document with a topic, which is identified by cluster representatives.

Usually, the cluster representatives are probabilistic models or cluster prototypes. However, in the document clustering, representatives such as the cluster prototype are not very useful to identify the topic addressed by the documents in each cluster. The document clusters are better identified by descriptors, which are terms present in the documents and significant to the topic addressed in the documents. Since documents are represented by a high dimensional feature space, the extraction of good descriptors is a challenging problem. The extraction of cluster descriptors is even more challenging in the flexible organization of documents using overlapping clustering, since

the same descriptor can be representative for more than one cluster with different weights of representativeness.

The task of extracting document cluster descriptors is usually divided in two kind of methods: Description Comes First (DCF) and Description Comes Last (DCL) [21]. By means of DCF methods, also known as label-based, the descriptors are extracted in a document preprocessing step before or at the same time of the document clustering. By means of DCL methods, also known as document-based, the descriptors are extracted after the document clustering.

According to [21], for DCF methods there is a "semantic interval" between the cluster descriptor extraction and the cluster prototypes, which contradicts the intuition "First clustering, second description", and decreases the explanatory ability of the cluster descriptor. On the other hand, DCL are typically less complex and capable of both good clustering performance and meaningful descriptors. In addition, by separating the clustering algorithm from the cluster descriptor extraction, a number of different algorithms can be tested and used.

We have proposed a DCL method to automatically discover overlapping cluster descriptors [22][23]. The proposed method extracts the best descriptors of a cluster from a rank of descriptor candidates. It can extract descriptors after the document clustering by means of FCM or PCM clustering algorithms in order to achieve the flexible organization of documents. However, the performance of each clustering algorithm can affect the extraction of meaningful overlapping cluster descriptors because the proposed method is dependent on the membership degrees of each document in each cluster. Therefore, in this paper we investigate whether or not well-known collections present different results when their document cluster descriptors are extracted after FCM and PCM. Such results are obtained by using document cluster descriptors as features for text categorization.

To present the proposed investigation, this paper is organized as follows. In Section II, basic concepts concerning flexible organization of documents and overlapping cluster descriptor extraction are reviewed. In Section III, the difference between the descriptors obtained after a fuzzy and a possibilistic clustering algorithm is presented. In Section IV, the experimental results concerning the performance of the descriptors extracted after FCM and PCM are presented, followed by discussions about the achieved results. Finally, in Section V, the conclusion and the future directions of this research is also presented.

II. Flexible organization of documents

In this section, we review the basic concepts and the methods used in our approach proposed in [22] and improved in [23] to organize documents in a flexible way.

A. Document preprocessing

The preprocessing of documents is necessary to structure the documents in order to make them processable by the algorithms of pattern extraction. The most common output of a document preprocessing is the representation of a document collection in a vector space in the form of a document-term matrix. Each matrix row corresponds to one document in the collection and each matrix column corresponds to one term in the entire collection of documents.

The terms in the document-term matrix are first examined in an initial effort to disregard terms that do not represent useful knowledge. In this step of examination, three tasks are very common: (1) Elimination of stopwords, which are words that are not relevant in the analysis of documents and usually consist of prepositions, pronouns, articles, interjections, among others; (2) Stemming, a technique that reduce the words to their root form in order to reduce the number of terms needed to represent the document collection; (3) n-gram extraction, which is the extraction of terms represented by n consecutive words, since words that occur in sequence in the document may contain more information than isolated words.

After selecting the terms that represent the document collection, for the proposed approach, the document-term matrix contains in its cells the ratio between the frequency of a particular term in a document and the inverse of the frequency of this term in the document collection (tf-idf Term Frequency-Inverse Document Frequency). By this measure, the importance of the terms in a document is weighted, so that terms which are present in a lot of documents have a smaller weight than the terms that occur more rarely in the collection.

The definition of tf-idf and preprocessed document-term matrix is presented next.

Definition 2.1: Let D be a document collection and d a general document in D. The frequency of a term t in document d, denoted by $tf(t,d)$, is the number of times that t occurs in d. The inverse of the frequency of the term t in the collection D is given by $idf(t) = log\frac{n}{d(t)}$, where n is the number of documents in D and $d(t)$ is the number of documents in D where t occurs. The measure $tf - idf$ (term frequency-inverse document frequency) of a term t in a document d from a colection D is defined as $tf - idf(t,d) = tf(t,d) \times idf(t,D)$.

Definition 2.2: Consider a document collection $D = \{\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_n\}$ and let T be the number of terms in the collection. A document-term matrix $W = [d_{kj}]$ is composed by a document $\mathbf{d}_k$ in each row such that each column corresponds to a term t_j, $j = 1, ..., T$. A document $\mathbf{d}_k$ is represented by a vector $[d_{k1}, d_{k2}, ..., d_{kT}]$, $1 \leq k \leq n$. This vector comprises the frequency of each term t in the document $\mathbf{d}_k$, weighted by how often this term occurs in the collection, i.e., $d_{kj} = tf - idf(t_j, d_k)$ (see Definition 2.1).

The document-term matrix is inherently high dimensional and sparse, which sometimes can make the document organization computationally very expensive or even impossible. This negatively affects the outcome of some knowledge extraction algorithms.

Therefore, to make the flexible organization of documents possible, the preprocessed documents are clustered by means of overlapping clustering algorithms described next.

B. Overlapping document clustering

The FCM[12] and PCM[13] clustering algorithms are important techniques to organize documents into clusters, since by these algorithms the documents can belong to more than one cluster with different membership degrees.

In order to carry out the flexible organization of documents, the documents are clustered by means of a slightly modified version of both clustering algorithms, FCM and PCM. The modification is related to the similarity measure between two documents, since the document-term matrix is sparse and has a high dimensionality. Such measure is defined as follows.

Definition 2.3: Let n be the number of documents in the collection and c be the number of clusters. Consider a document $\mathbf{d}_k$, $k = 1, ..., n$, and a cluster prototype $\mathbf{v}_i$, $i = 1, ..., c$. The dissimilarity between a document and a prototype $\|\mathbf{d}_k - \mathbf{v}_i\|$ is measured using the cosine coefficient similarity according to Equation(1) and Equation (2).

$$sim(\mathbf{d}_k, \mathbf{v}_i) = cos\theta = \frac{\mathbf{d}_k \cdot \mathbf{v}_i}{|\mathbf{d}_k| \, |\mathbf{v}_i|} \in [0, 1] \tag{1}$$

$$\|\mathbf{d}_k - \mathbf{v}_i\| = 1 - sim(\mathbf{d}_k, \mathbf{v}_i) \in [0, 1] \tag{2}$$

Although by means of both FCM and PCM algorithms it is possible to obtain the compatibility of documents in more than one cluster, they represent different concepts of overlapping, which influence the compatibility degrees found in their clustering process. Therefore, each algorithm has its particularities and is performed as follows.

1) Fuzzy C-Means: The FCM [12] algorithm used for document clustering is an iterative process that updates the prototypes of the clusters defined initially from a fuzzy pseudo-partition and the partition matrix giving the membership degree of each document to each cluster. This update tries to minimize the dissimilarity between a document and a cluster prototype. The pseudo-partition is defined as follows [24].

Definition 2.4: Let c be the number of clusters and $A_i(\mathbf{d}_k)$ the membership degree of the document $\mathbf{d}_k$ in the cluster i, $k = 1, ..., n$, $i = 1, ..., c$. A fuzzy pseudo-partition $U = [A_i(\mathbf{d}_k)]$ is a family of fuzzy sets of D (see Definition 2.2) denoted by $\mathrm{P} = \{A_1, A_2, ..., A_c\}$, which satisfies the Equations (3) and (4).

$$\sum_{i=1}^{c} A_i(\mathbf{d}_k) = 1 \tag{3}$$

$$0 < \sum_{k=1}^{n} A_i(\mathbf{d}_k) < n \tag{4}$$

During the clustering procedure, the prototypes and the partition matrix are updated until a stopping criterion is satisfied. Let n be the number of documents in the collection and c be the number of clusters. The document cluster prototypes $V = \{\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_c\}$ are calculated according to Equation (5), where $m > 1$ is a real number, called fuzzification factor, that controls the influence of the membership degrees in the fuzzy clustering. In the experiments presented in this paper, $m = 2.5$ was used for clustering all datasets.

$$\boldsymbol{v}_i = \frac{\sum_{k=1}^{n} [A_i(\boldsymbol{d}_k)]^m \boldsymbol{d}_k}{\sum_{k=1}^{n} [A_i(\boldsymbol{d}_k)]^m}, \; i = 1, ..., c, k = 1, ..., n. \tag{5}$$

Further, based on Equation (5) and the definition of dissimilarity presented in Definition 2.3, the FCM algorithm updates the fuzzy pseudo-partition according to Equation (6).

$$\mu_{ik} = A_i(\boldsymbol{d}_k) = \frac{1}{\sum_{j=1}^{c} \left(\frac{\|\boldsymbol{d}_k - \boldsymbol{v}_i\|}{\|\boldsymbol{d}_k - \boldsymbol{v}_j\|} \right)^{\frac{1}{m-1}}} \tag{6}$$

The goal of FCM is to minimize the optimization function J_m, defined in Equation (7). The performance of FCM is based on the J_m optimization under the fuzzy pseudo-partition U defined in Definition 2.4.

$$J_m(U) = \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ik}^m \|\boldsymbol{d}_k - \boldsymbol{v}_i\| \tag{7}$$

Furthermore, before FCM starts running, the number of groups c, a small number ϵ as stopping criteria and a fuzzification factor m must be defined.

2) Possibilistic C-Means: The membership degree $A_i(\boldsymbol{d}_k)$ that FCM assigns to a document $\boldsymbol{d}_k$ is related to the relative distance of $\boldsymbol{d}_k$ to the cluster prototype $\boldsymbol{v}_i$, $i = 1, ..., c$. If $\mathbf{d}_k$ is equidistant to two prototypes, $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, the membership degree of $\boldsymbol{d}_k$ in each cluster will be the same: $A_1(\boldsymbol{d}_k) = 0.5$ and $A_2(\mathbf{d}_k) = 0.5$.

Let us consider a noise data as a document that is far but equidistant from the prototypes of two clusters. By means of FCM, noise data can be assigned to both clusters with the same membership degrees as the documents closer and also equidistant to the cluster prototypes. In Figure 1 we illustrate such situation, in which $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ have both the same membership degree, 0.5, in the clusters A_1 and A_2, although document $\boldsymbol{d}_1$ is closer to the clusters prototypes than $\boldsymbol{d}_2$.

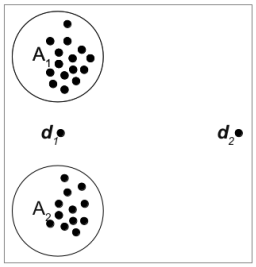

Fig. 1. Position of documents $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ related to clusters A_1 and A_2 (Adapted from [25])

According to Pal et al. in [26], such a situation illustrate the basic notion of probabilistic partitioning of data sets of FCM, which has the constraint $\sum_{i=1}^{c} A_i(\boldsymbol{d}_k) = 1$, i.e, the sum of a document membership degree in all clusters must be equal

to 1. Therefore, the PCM algorithm was developed to relax this constraint of FCM, considering the absolute value of the distance of $\boldsymbol{d}_k$ from the cluster prototypes. Considering such looseness, the $A_i(\boldsymbol{d}_k)$ obtained by means of PCM should be interpreted as the typicality of a document $\boldsymbol{d}_k$ relative to cluster i.

In a similar way as FCM, the PCM [13] algorithm used for document clustering is an iterative process that updates the prototypes of the clusters defined initially from a pseudo-partition (degrees of typicality of every object in all clusters). This update tries to minimize the dissimilarity between a document and a cluster prototype. Its cluster prototype update is identical to that in Equation (5). Further, based on the definition of dissimilarity presented in Definition 2.3, the PCM updates the pseudo-partition according to Equation (8) [26].

$$\sigma_{ik} = A_i(\boldsymbol{d}_k) = \frac{1}{1 + \left(\frac{\|\boldsymbol{d}_k - \boldsymbol{v}_i\|}{\gamma_i} \right)^{\frac{1}{m-1}}} \tag{8}$$

The user-defined constant $\gamma_i > 0$ is considered to minimize the singularity problem of FCM. Therefore, the distance $\|\boldsymbol{d}_k - \boldsymbol{v}_i\|$ can be zero, relaxing the constraint in Equation (3). As recommended by Krishnapuram and Keller in [13], we have chosen γ_i according to Equation (9).

$$\gamma_i = \frac{\sum_{k=1}^{n} \mu_{ik}^m \|\boldsymbol{d}_k - \boldsymbol{v}_i\|}{\sum_{k=1}^{n} \mu_{ik}^m} \tag{9}$$

where μ_{ik} is a terminal FCM partition of D according to Equation (6).

The goal of PCM is to minimize the optimization function P_m, according to Equation (10) with the tipicalities matrix $H = [\sigma_{ik}]$.

$$P_m(H) = \sum_{k=1}^{n} \sum_{i=1}^{c} {\sigma_{ik}}^m \|\boldsymbol{d}_k - \boldsymbol{v}_i\| + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{n} (1 - \sigma_{ik})^m \tag{10}$$

The first term of P_m is just the function J_m, and in the absence of the second term, unconstrained optimization will lead to the trivial solution $\sigma_{ik} = 0$, $i = 1, ..., c$, $k = 1, ..., n$. The second term of P_m acts as a penalty which tries to bring σ_{ik} toward 1 [26].

Finally, once the documents are clustered, the overlapping cluster descriptors can be extracted by the method described next.

C. Overlapping cluster descriptor extraction

The method proposed in [22] carries out a procedure that uses an adaptation of the classical measures of information retrieval [27] namely precision, recall, and $f1$-measure, which is the weighted harmonic mean of precision and recall.

In the fuzzy and possibilistic cluster descriptor extraction, all the terms found in the document preprocessing step are initially considered as descriptor candidates. Additionally, a document $\mathbf{d}_k$ is considered to belong to cluster i if it has a membership degree $A_i(\mathbf{d}_k) \geq s$, where $s = \frac{1}{c}$. The threshold s is considered for two reasons. Firstly, its use allows the selection of descriptor candidates from documents that belong to more than one cluster with different compatibility degrees, instead of considering only the cluster with the highest compatibility degree. Secondly, using this threshold it is possible to penalize the descriptor candidates that occur in documents with low compatibility degree in a cluster.

A rank of terms weighted by their $f1$-measure is obtained for each cluster as follows, considering the contingency matrix presented in Table I.

TABLE I. CONTINGENCY MATRIX FOR INFORMATION RETRIEVAL MEASUREMENT

	Documents of cluster c (documents with compatibility degree in cluster c higher than or equal to s)	Documents that are not in cluster c (documents with compatibility degree in cluster c lower than s)
Documents which have the descriptor candidate t	*hits*	*noises*
Documents which do not have the descriptor candidate t	*losses*	*rejects*

i) Calculate the precision of a descriptor candidate t in a cluster c:

$$p(t, c) = \frac{hits}{hits + noises} \tag{11}$$

ii) Calculate the recall of a descriptor candidate t in a cluster c:

$$r(t, c) = \frac{hits}{hits + losses} \tag{12}$$

iii) Calculate the $f1$-measure of a descriptor candidate t in a cluster c:

$$f1(t, c) = \frac{2 \cdot p(t, c) \cdot r(t, c)}{p(t, c) + r(t, c)} \tag{13}$$

Since the ranking of descriptor candidates is obtained, the descriptors are selected. The number of descriptors to be selected depends on the application.

The difference between the descriptors obtained from a fuzzy and a possibilistic approach is presented next.

III. COMPARISON OF FUZZY AND POSSIBILISTIC DESCRIPTORS

To check whether or not the compatibility degree influences the extraction of overlapped cluster descriptors, let us consider the situation in which there are 3 documents composed by 3 terms each one. Such documents were clustered in 2 groups and the cluster descriptors for each cluster were extracted using FCM and PCM.

To extract the cluster descriptors consider the document-term matrix presented in Table II and the document-group matrix presented in Table III, where the compatibility degrees were obtained using FCM.

Each term presented in Table II is considered a descriptor candidate. The extraction of the descriptors of a particular

TABLE II. EXAMPLE DOCUMENT-TERM MATRIX

	$\mathbf{t}_1$	$\mathbf{t}_2$	$\mathbf{t}_3$
$\mathbf{d}_1$	0	0	1
$\mathbf{d}_2$	1	1	1
$\mathbf{d}_3$	0	1	1

TABLE III. EXAMPLE DOCUMENT-GROUP MATRIX (MEMBERSHIP DEGREES)

	$\mathbf{A}_1$	$\mathbf{A}_2$
$\mathbf{d}_1$	0.5	0.5
$\mathbf{d}_2$	0.3	0.7
$\mathbf{d}_3$	0.6	0.4

cluster begins with the calculation of the $f1$-measure of each descriptor candidate.

From this measurement, we found the relevance of each descriptor for each cluster by means of Equations (11), (12) and (13): $f_1(t_1, A_1) = 0$, $f_1(t_2, A_1) = 0.5$, $f_1(t_3, A_1) = 0.79$, $f_1(t_1, A_2) = 0.5$, $f_1(t_2, A_2) = 0.66$, $f_1(t_3, A_2) = 1.0$.

We have also obtained compatibility degrees obtained using PCM, presented in Table IV.

TABLE IV. EXAMPLE DOCUMENT-GROUP MATRIX (TYPICALITY DEGREES)

	$\mathbf{A}_1$	$\mathbf{A}_2$
$\mathbf{d}_1$	0.3	0.3
$\mathbf{d}_2$	0.3	0.7
$\mathbf{d}_n$	0.6	0.4

However, for the same documents of the example, we found a different relevance for the cluster descriptor candidates by means of Equations (11), (12) and (13): $f_1(t_1, A_1) = 0$, $f_1(t_2, A_1) = 1.0$, $f_1(t_3, A_1) = 0.49$, $f_1(t_1, A_2) = 1.0$, $f_1(t_2, A_2) = 0.66$, $f_1(t_3, A_2) = 0.49$.

From the example, we obtained a different rank from each clustering algorithm. Using FCM, the rank of descriptors for cluster 1 is $t_1 < t_2 < t_3$, and for cluster 2 is $t_1 < t_2 < t_3$. Using PCM, the rank of descriptors for cluster 1 is $t_1 < t_3 < t_2$, and for cluster 2 is $t_3 < t_2 < t_1$.

Since the clusters descriptors are obtained, the flexible organization of documents can be formally defined. Each cluster is identified by a topic, which is the set of descriptors extracted for that cluster. The definition of topic is presented next.

Definition 3.1: Let $P = \{A_1, A_2, ..., A_c\}$, where c is the number of clusters, be the pseudo-partition resulting from the clustering of a document collection D. The topic t_i, associated with cluster A_i, where $i = 1, ..., c$, is a set of descriptors extracted from A_i as defined in Section II-C. The clustering P is identified by a set of topics $T = \{t_1, t_2, ..., t_c\}$.

The flexible organization by means of fuzzy and possibilistic document clustering are defined in Definition 3.2 and 3.3, respectively.

Definition 3.2: Let D be a collection of documents. The fuzzy flexible organization is a pair $F = (U, T)$, where U is the fuzzy document-cluster matrix found by means of FCM from the collection D and T is the set of topics associated to the partition defined by U.

Definition 3.3: Let D be a collection of documents. The possibilistic flexible organization of D is a pair $P = (H, T)$ where H is the possibilistic document-cluster matrix found by means of PCM from D and T is the set of topics associated to the partition defined by H.

The document organization by means of cluster descriptors can be affected by the difference in the rankings obtained by each clustering algorithm, as illustrated by the previous discussion. To check if the difference of ranking obtained from FCM and PCM affects the quality of the descriptors obtained, we have performed some experiments using different document collections.

IV. EXPERIMENTAL RESULTS

In general, the quality of cluster descriptors are measured by the performance of the clustering algorithm. However, when the organization of documents is achieved using document clusters, the quality of the cluster descriptors should be evaluated considering their conciseness, which means that they should be as short as possible, but sufficient enough to convey the topic of the cluster; their comprehensibility, also known as transparency, which means that they should map the content of the clusters; accuracy, which means that they should reflect the topic of the corresponding cluster; and, distinctiveness, which means that they are more frequent on one cluster than in others [28].

A recently published fuzzy DCL method named Fuzzy Transduction-based Clustering Algorithm (FTCA) [6] was developed to cluster similar documents and facilitate a term frequency based descriptor extraction. Although FTCA presents good improvements related to two common DCF methods (Suffix Tree Clustering (STC) [29] and Lingo [30]), it was evaluated just concerning the clustering algorithm performance. Moreover, STC and Lingo are crisp methods.

The method proposed in [22], and reviewed in Section II, is a DCL method. But different of the FTCA, it ranks the descriptor candidates using some measures from the Information Retrieval field improving the flexible organization of documents.

After clustering the documents by means of FCM and PCM, we extract fuzzy cluster descriptors from the obtained clusters. To evaluate the performance of these descriptors concerning their comprehensibility, we checked the power prediction of the descriptors in the sense that they can be used as good attributes for text categorization. For this, we performed well-known classification algorithms of machine learning: SVM, Naive Bayes, Multinomial Naive Bayes, KNN and C4.5.

We evaluated the predictive power of the descriptors considering each cluster as a class and the descriptors as document attributes. Since in fuzzy clustering the documents can belong to more than one cluster, the document class is the cluster in which it has the highest membership degree. After labeling each document in the collection with the corresponding cluster, an attribute-value matrix was created with each descriptor being an attribute. The matrix entries are the frequency of the descriptors in each document.

We carried out these experiments using six different document collections, whose general characteristics are summarized in Table V. To ensure diversity in the collections, they were obtained from different sources. All collections were preprocessed using the Pretext[1] tool [31]. Any term that occurs in fewer than two documents was eliminated and 1-gram terms were selected, i.e, terms composed of one word.

TABLE V. DOCUMENT COLLECTIONS

Dataset	# classes	# docs
Opinosis	3	51
20Newsgroups	4	2000
Hitech	6	600
NSF	16	1600
WAP	20	1560
Reuters-21578	43	1052

The Opinosis collection [32] contains documents composed by customer reviews about characteristics of some products. The customer reviews were obtained from the websites: Tripadvisor.com, Amazon.com and Edmunds.com, which provide customer reviews about hotels, cars and electronics products, respectively. The Opinosis collection is available at UCI Machine Learning Repository [33].

The 20Newsgroups collection has become a popular data set for experiments in text applications of machine learning techniques, such as text classification and text clustering. It was collected by Lang for the Newsweeder research presented in [34]. The original 20Newsgroups collection has approximately 20000 newsgroup documents, partitioned (nearly) in 20 different newsgroups. For the experiments carried out for the work presented here, we selected the documents from the category science, which is composed by the newsgroups sci.crypt, sci.electronics, sci.med and sci.space. The subset of 20Newsgroups collection has 2000 documents.

The NSF collection was downloaded from the UCI Machine Learning Repository [33]. The original collection consists of 129000 abstracts describing NSF (National Science Foundation) awards for basic research. We have selected 1600 for our experiments.

The Hitech collection comes from the Text REtrieval Conference (TREC)[2]. The documents are composed of newspaper stories from the San Jose Mercury News classified into different topics. The original collection consists of 2301 documents. We have selected 600 documents for our experiments.

The Wap collection was obtained by Moore *et al.* for the WebACE project [35]. Each document corresponds to a web page listed in the subject hierarchy of Yahoo![3]. The original collection composed by 1560 distributed in 20 categories was used.

The Reuters-21578[4] is one of the most used test collection for text categorization research. The collection was obtained by the Carnegie Group, Inc. and Reuters, Ltd. in the course of developing the CONSTRUE text categorization system [36]. This collection is composed by 21578 documents in its original format. We selected 43 categories in a total of 1052 documents.

[1]http://sites.labic.icmc.usp.br/pretext2/

[2]http://trec.nist.gov

[3]http://www.yahoo.com

[4]http://www.daviddlewis.com/resources/testcollections/

The classification was carried out using the WEKA tool [37]. The Naive Bayes (NB), Multinomial Naive Bayes (NB-Multinominal) and J48 algorithms (the weka implementation of the C4.5 classification method) were executed using the default parameters of the tool. However, the performance of the SVM was tuned up using the Normalized Polynomial Kernel and the complexity parameter c=2.0. The IBk (the weka implementation of the KNN classification method) was experimented ranging the number of neighbors from 1 to 7. The best result was obtained using 5 neighbors.

The 10-fold cross validation method was used in all experiments. The number of clusters for FCM and PCM algorithms was defined using the number of classes of each dataset as input for the Fuzzy Silhouette (FS) [38] method. Such a method is commonly used to evaluate document clustering and choose the best number of clusters for the document organization. Therefore, the number of clusters is determined considering the best value of silhuette obtained from a number of clusters between 2 and the number of classes of each dataset. In situations where the number of classes are unknown, the number of clusters can also be determined by using the FS, however, the maximum number of clusters need to be defined empirically.

The performance rates (correct classification rate) obtained from each classifier over each document collection are presented in Figures 2, 3, 4, 5, 6, and 7.

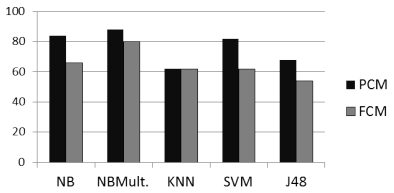

Fig. 2. Performance of cluster descriptors obtained from PCM and FCM clustering algorithms (Opinosis collection)

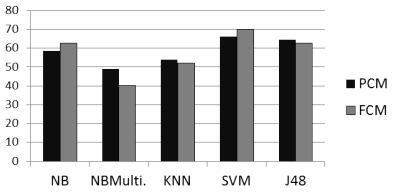

Fig. 3. Performance of cluster descriptors obtained from PCM and FCM clustering algorithms (20Newsgroups collection)

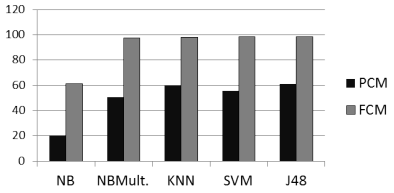

Fig. 4. Performance of cluster descriptors obtained from PCM and FCM clustering algorithms (Reuters-21578 collection)

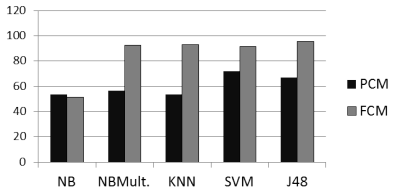

Fig. 5. Performance of cluster descriptors obtained from PCM and FCM clustering algorithms (WAP collection)

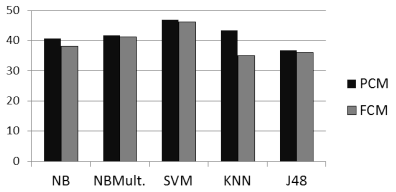

Fig. 6. Performance of cluster descriptors obtained from PCM and FCM clustering algorithms (Hitech collection)

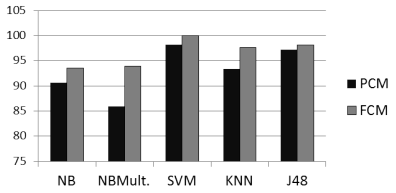

Fig. 7. Performance of cluster descriptors obtained from PCM and FCM clustering algorithms (NSF collection)

In summary, the performance of the descriptors obtained after clustering by means of FCM and PCM is presented in Table VI. The check mark (✓) shows from which clustering method the descriptors of each collection have obtained the best results.

TABLE VI. PERFORMANCE OF FCM AND PCM OVER ALL THE COLLECTIONS

Dataset	FCM	PCM
Opinosis		✓
20Newsgroups		✓
Hitech		✓
NSF	✓	
WAP	✓	
Reuters	✓	

According to the results, the descriptors obtained from collections with a small number of classes have a better performance when extracted after the clustering by means of PCM than FCM. Although the number of classes do not correspond to the number of clusters, such result is due to the fact that the PCM presents the problem of coincident clusters.

The problem of coincident clusters occurs when the initialization of the clustering concerning the typicality matrix is not sufficiently distinct, i. e, the clustering algorithm results in c clusters, although the correct number of clusters is c', where $c' < c$. When a collection presents many classes of documents, the vector that composes a document is usually sparse, since documents of different classes are composed by different terms. This characteristic complicates the start up of the possibilistic clustering algorithm.

To avoid the initialization problem, the PCM algorithm was initialized using terminal outputs of FCM, as suggested by Krishnapuram and Keller in [13]. However, the results obtained show that there is no guarantee that $c' = c$ is the best number of clusters even when the results from the Fuzzy Silhouette suggest this [26].

Another reason for the obtained results is that FCM has more chance to obtain a good membership degree for a document considered a noise, since it does a probabilistic distribution of membership degrees over the documents. On the other hand, PCM has a higher sensitivity to noise data than FCM, considerably lowering the value of typicality.

Collections with higher number of classes are more likely to have noise data, leading PCM to find lower values of typicality. In this case, some descriptors may not be selected for a particular group, since these descriptors are typical of a document considered a noise.

V. CONCLUSION

This paper presented a comparative study concerning the performance of cluster descriptors obtained after fuzzy and possibilistic clustering algorithms for flexible document organization.

The experiments showed that the method for extracting cluster descriptors is a promising method in the sense that the descriptors can be used as good attributes for text categorization. However, the method is dependent on the membership degrees of each document in each cluster obtained by an overlapping clustering algorithm.

Therefore, we have investigated whether or not well-known collections present different results when their document cluster descriptors are extracted after two clustering algorithms, FCM and PCM.

From this investigation, we conclude that each algorithm should be used depending on the collection characteristics. Collections composed by a large number of classes have better results when the cluster descriptors are extracted after the FCM in stead of after PCM. This is due to the fact that the PCM presents the problem of coincident clusters which is increased by the big number of classes. At the same time, the PCM is a good alternative when the collection presents noise data.

It is known that a collection of documents is often a mixture of relevant and irrelevant documents. A balance between noise data identification and classification results is the key for the choice of a good clustering algorithm and, consequently, meaningful cluster descriptors extraction.

As future work, we intent to investigate effective techniques by comparing the PCM and FCM results when considering some documents known as relevant in a semi-supervised scenario. Therefore, it is expected that more meaningful cluster descriptors are to be extracted after PCM and FCM semi-supervised clustering algorithms.

Moreover, comparative analysis will be conducted comparing the already obtained results with results obtained from the possibilistic fuzzy c-means (PFCM) clustering algorithm. The PFCM is a hybridization of possibilistic c-means (PCM) and fuzzy c-means (FCM) that often avoids various problems of

PCM and FCM, such as the noise sensitivity defect of FCM and the coincident clusters problem of PCM.

ACKNOWLEDGMENT

Authors acknowledge the Brazilian research agency CAPES (Coordination for the Improvement of Higher Level Personnel) for their support with the PDSE grant 5983-11-8. This paper is also based upon work supported by FAPESP (Sao Paulo Research Foundation), Brazil, under the grant 2011/19850-9.

REFERENCES

[1] E. Herrera-Viedma, "Modeling the retrieval process for an information retrieval system using an ordinal fuzzy linguistic approach," *Journal of the American Society for Information Science and Technology*, vol. 52, no. 6, pp. 460–475, 2001.

[2] C. Chowdhury and P. Bhuyan, "Information retrieval using fuzzy c-means clustering and modified vector space model," in *3rd IEEE International Conference on Computer Science and Information Technology*, vol. 1, 2010, pp. 696–700.

[3] D. H. Kraft, G. Pasi, and G. Bordogna, "Vagueness and uncertainty in information retrieval: How can fuzzy sets help?" *Proceedings of the 2006 International Workshop on Research Issues in Digital Libraries*, pp. 1–10, 2006.

[4] G. Bordogna and G. Pasi, "Soft clustering for information retrieval applications," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 2, pp. 138–146, 2011.

[5] M. S. Haghighi, H. S. Yazdi, and A. Vahedian, "A hierarchical possibilistic clustering," *International Journal of Computer Theory and Engineering*, vol. 1, no. 4, pp. 465–472, 2009.

[6] T. Matsumoto and E. Hung, "A transduction-based approach to fuzzy clustering, relevance ranking and cluster label generation on web search results," *Journal of Intelligent Information Systems*, vol. 38, no. 2, pp. 419–448, 2012.

[7] A. Garcia-Plaza, V. Fresno, and R. Martinez, "Fitting document representation to specific datasets by adjusting membership functions," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, June 2012, pp. 1–8.

[8] W. L. Chang, K. M. Tay, and C. P. Lim, "Enhancing an evolving tree-based text document visualization model with fuzzy c-means clustering," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, July 2013, pp. 1–6.

[9] Y. Wang, L. Chen, and J.-P. Mei, "Stochastic gradient descent based fuzzy clustering for large data," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, July 2014, pp. 2511–2518.

[10] K. Honda, D. Tanaka, and A. Notsu, "Incremental algorithms for fuzzy co-clustering of very large cooccurrence matrix," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, July 2014, pp. 2494–2499.

[11] P. Fazendeiro and J. V. Oliveira, "Observer-biased fuzzy clustering," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 85–95, 2015.

[12] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. Norwell, MA, USA: Kluwer Academic Publishers, 1981.

[13] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 2, pp. 98–110, 1993.

[14] W.-C. Tjhi and L. Chen, "Possibilistic fuzzy co-clustering of large document collections," *Pattern Recognition*, vol. 40, pp. 3452–3466, 2007.

[15] M. Boughanem, H. Prade, and O. Bouidghaghen, "Extracting topics in texts: Towards a fuzzy logic approach," in *Proceedings of the Information Processing and Management of Uncertainty (IPMU)*, 2008, pp. 1733–1740.

[16] R. Saracoglu, K. TuTuncu, and N. Allahverdi, "A new approach on search for similar documents with multiple categories using fuzzy clustering," *Expert Systems with Applications*, vol. 34, pp. 2545–2554, 2008.

[17] W.-C. Tjhi and L. Chen, "Dual fuzzy-possibilistic coclustering for categorization of documents," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 3, pp. 532–543, 2009.

[18] H. Jiang, F. Ye, J. Gu, Y. Liu, M. Zhu, and D. Chen, "An improved method of fuzzy clustering algorithm and its application in text clustering," *Journal of Information & Computational Science*, vol. 10, no. 2, pp. 519–526, 2013.

[19] J.-P. Mei, Y. Wang, L. Chen, and C. Miao, "Incremental fuzzy clustering for document categorization," in *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, July 2014, pp. 1518–1525.

[20] C. Zhang, "Document clustering description based on combination strategy," in *Proceedings of the International Conference on Innovative Computing,Information and Control*, vol. 0, 2009, pp. 1084–1088.

[21] T. Nogueira, S. Rezende, and H. Camargo, "Fuzzy cluster descriptor extraction for flexible organization of documents," in *Proceedings of the 11th International Conference on Hybrid Intelligent Systems (HIS)*, 2011, pp. 528–533.

[22] T. Nogueira, H. Camargo, and S. Rezende, "Fuzzy cluster descriptors improve flexible organization of documents," in *Proceedings of the 12th International Conference on Intelligent Systems Design and Applications (ISDA)*, 2012, pp. 616–621.

[23] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: theory and applications*, 1st ed. Prentice-Hall, 1995.

[24] J. V. d. Oliveira and W. Pedrycz, *Advances in Fuzzy Clustering and its Applications*. New York, NY, USA: John Wiley & Sons, Inc., 2007.

[25] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek, "A possibilistic fuzzy c-means clustering algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 4, pp. 517–530, 2005.

[26] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc., 1986.

[27] C. Zhang, H. Wang, Y. Liu, and H. Xu, "Document clustering description extraction and its application," in *Proceedings of the 22nd International Conference on Computer Processing of Oriental Languages. Language Technology for the Knowledge-based Economy*, 2009, pp. 370–377.

[28] O. Zamir and O. Etzioni, "Web document clustering: a feasibility demonstration," in *Proceedings of the 21st annual international ACM SIGIR (Special Interest Group on Information Retrieval) conference on Research and development in information retrieval*, 1998, pp. 46–54.

[29] S. Osinski and D. Weiss, "A concept-driven algorithm for clustering search results," *IEEE Intelligent Systems*, vol. 20, no. 3, pp. 48–54, 2005.

[30] M. V. B. Soares, R. C. Prati, and M. C. Monard, "PRETEXT II: Description of restructuring tool preprocessing of texts," ICMC-USP, Tech. Rep. 333, 2008, (in portuguese).

[31] K. Ganesan, C. Zhai, and J. Han, "Opinosis: A graph based approach to abstractive summarization of highly redundant opinions," in *Proceedings of the 23rd International Conference on Computational Linguistics (COLING), 2010*, 2010, pp. 340–348.

[32] A. Frank and A. Asuncion, "UCI machine learning repository," 2010. [Online]. Available: [http://archive.ics.uci.edu/ml]

[33] K. Lang, "Newsweeder: Learning to filter netnews," in *Proceedings of the Twelfth International Conference on Machine Learning*, 1995, pp. 331–339.

[34] E.-H. Han, D. Boley, M. Gini, R. Gross, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore, "Webace: a web agent for document categorization and exploration," in *Proceedings of the second international conference on Autonomous agents*, ser. AGENTS '98. New York, NY, USA: ACM, 1998, pp. 408–415. [Online]. Available: http://doi.acm.org/10.1145/280765.280872

[35] P. J. Hayes and S. P. Weinstein, "Construe/TIS: A system for content-based indexing of a database of news stories." in *2nd Annual Conference on Innovative Applications of Artificial Intelligence.*, 1990, pp. 1–5.

[36] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explorations*, vol. 11, no. 1, pp. 10–18, Nov. 2009.

[37] R. Campello and E. Hruschka, "A fuzzy extension of the silhouette width criterion for cluster analysis," *Fuzzy Sets and Systems*, vol. 157, no. 21, pp. 2858 – 2875, 2006.