A Method to Generate Soft Reference Data for Topic Identification

Daniel Vélez[1], Guillermo Villarino[2], J. Tinguaro Rodríguez[1,3(✉)], and Daniel Gómez[2]

[1] Faculty of Mathematics, Complutense University of Madrid, Madrid, Spain
{danielvelezserrano,jtrodrig}@mat.ucm.es
[2] Faculty of Statistical Studies, Complutense University of Madrid, Madrid, Spain
{gvillari,dagomez}@estad.ucm.es
[3] Interdisciplinary Mathematics Institute, Complutense University of Madrid, Madrid, Spain

Abstract. Text mining and topic identification models are becoming increasingly relevant to extract value from the huge amount of unstructured textual information that companies obtain from their users and clients nowadays. Soft approaches to these problems are also gaining relevance, as in some contexts it may be unrealistic to assume that any document has to be associated to a single topic without any further consideration of the involved uncertainties. However, there is an almost total lack of reference documents allowing a proper assessment of the performance of soft classifiers in such soft topic identification tasks. To address this lack, in this paper a method is proposed that generates topic identification reference documents with a soft but objective nature, and which proceeds by combining, in random but known proportions, phrases of existing documents dealing with different topics. We also provide a computational study illustrating the application of the proposed method on a well-known benchmark for topic identification, as well as showing the possibility of carrying out an informative evaluation of soft classifiers in the context of soft topic identification.

Keywords: Soft classification · Text mining · Topic identification

1 Introduction

In recent years, there has been a significant growth in the volume of available data for companies from different industries regarding their clients. With the aim of being able to exploit such information, the application of mathematical and machine learning methods allowing to identify patterns and relationships useful for decision making purposes has also known an important proliferation.

Frequently, data regarding or coming from clients have an unstructured nature, and particularly it is estimated that approximately 80% of that information is in textual form [1]. Consequently, the analytical field denominated as Text Mining, a multidisciplinary area based on natural language processing (NLP) and machine

Supported by the Government of Spain (grant PGC2018-096509-B-I00) and Complutense University of Madrid (UCM Research Group 910149).

M.-J. Lesot et al. (Eds.): IPMU 2020, CCIS 1239, pp. 54–67, 2020.
https://doi.org/10.1007/978-3-030-50153-2_5

learning techniques focused on extracting value from such texts, has experienced a considerable development. In this field, solutions oriented towards the identification of topics in texts have specially gained relevance, as they enable useful applications for the analysis of contents in social media (Facebook, Twitter, blogs, etc.) or document classification (spam filtering, sentiment analysis, customer complaint classification, etc.) [2].

It is important to note that the manual supervision process of a whole document corpus can be quite costly, which makes the employment of methods allowing to automatically label the texts highly convenient. To some extent, it is possible to manually label just a sample of the available documents, and then apply a supervised method to classify the remaining texts in the corpus from that training sample. However, this forces these new documents to be classified in some of the categories found in the supervised sample, even when their contents do not actually fit into any of them. This fact motivates that unsupervised methods are frequently applied instead of supervised ones.

Nevertheless, even when unsupervised methods are used, it is useful to have some labelled texts available in order to allow the assessment of their performance. In this way, for instance, it is usual to omit at first any knowledge about the topics of the labelled documents, then obtain the topic clusters, and finally validate whether the words that characterize the documents of each cluster are related to the known topics.

Another remark is that it is not realistic to assume that any document has to be always associated to a single topic. To address this situation, many soft classification techniques exist that allow simultaneously assigning a document to a set of topics up to different degrees. However, we have found an important lack regarding the assessment of the performance of such methods, as soft reference corpus for topic identification do not exist or are hardly available. For this reason, in this paper we propose a method to generate topic identification reference documents with a soft nature.

Specifically, the proposed method generates texts that combine phrases of previously available texts associated to different topics. The result is then a corpus of new documents such that, for each of these new documents, soft reference values are provided informing of the relative proportion of phrases it contains from each of the combined topics.

To illustrate the application of the proposed method and the possibilities it enables, a computational study based on a well-known benchmark corpus in topic identification is also included in this work.

This paper is structured as follows. Section 2 is devoted to discuss with more detail the need for soft reference datasets. The proposed method to generate soft reference corpuses for topic identification is then described in Sect. 3, and the computational study illustrating its usage is presented in Sect. 4. Finally, some conclusions and future work are provided in Sect. 5.

2 The Need for Soft Reference Data In topic Identification

The need for soft reference data in some classification contexts has since long been established. In such contexts, the very nature of the classification variable to

predict is soft, in such a way that objects to be classified can naturally belong or be assigned to different classes, simultaneously and up to a degree. For instance, in the field of land cover estimation at sub-pixel level, the objects to be classified are pixels in a terrain image, and the classes to be assigned are different terrain types, such as water, wetland, forest, and so on. For many years, the usual image classification approach to this problem consisted in assigning just one class, i.e. a single terrain type, to each pixel [3,4]. However, due to the limited resolution of the images, a pixel may cover an area of several squared meters, in which different terrain types can coexist. For this reason, assigning each image pixel to a single type of cover can be misleading, particularly in applications in which a more precise assessment of land cover is needed, or whenever the spatial resolution of pixels is rather low. In either case, it is important to notice that this kind of crisp assignment omits the complex nature of the problem, and can provide a false appearance of lack of uncertainty regarding the mix of terrain types actually occurring at pixel level.

One could think that a multilabel or label ranking model, respectively associating each pixel to a set or ranking of terrain types, should alleviate this problematic. However, these kinds of solution may be again insufficient, as neither one can fully represent the involved uncertainty regarding the composition of the land cover within each pixel. For instance, if the area associated to a given pixel is composed of a 70% water and a 30% wetland, a label ranking output given by the ordered pair $(water, wetland)$ just informs that *water* is more predominant at this area than *wetland*, but fails to adequately inform about the relative abundance of each cover. Of course, a multilabel (non-ordered) output of the type $\{water, wetland\}$ would be even less informative.

Rather, as this example illustrates, the appropriate, most informative representation estimates the proportion of the pixel area that is covered by each terrain type. Thus, it is a soft output the one that actually allows representing and managing the uncertainty associated to the land cover estimation problem. For this reason, the standard approach to address this problem gradually switched from crisp image classification to linear mixture models [5], and later to more accurate soft classification schemes, as e.g. fuzzy unsupervised and supervised classifiers or neural networks [6,7].

However, it is important to remark that the successful application of soft techniques in the land cover estimation context crucially depended on the availability of contextual soft reference data. Even when a soft supervised approach can be developed without an explicitly soft reference, the proper evaluation of both supervised and unsupervised techniques would have been impossible without adequate soft reference data.

This is also our point concerning some currently developing tasks in text mining, and particularly in topic identification. For instance, this last field is finding an increasing application in the analysis of customer complaint texts. The automatic analysis and understanding of complaint forms is becoming more and more relevant for companies in order to carry out adequate actions to improve their services. Within this context, the usual aim is to classify complaints according to their

causes from the basis of the text written by the customers at the complaint forms [8].

It is important to notice the link between this complaint causes classification problem and the previous land cover estimation problem. Particularly, it is possible that a complaint text (object to be classified) refers to more than one cause (classes to be assigned) simultaneously, and each of these causes can have a different weight within the text of a single complaint form. That is, it may be convenient to model the complaint classification problem as a soft classification problem, similarly to the land cover estimation problem described above.

Let us at this point introduce some notation to formalize in a general way the notions being discussed. Let X denote the set of objects to be classified, and let k denote the number of classes being considered. Then, in a soft classification framework each object $x \in X$ has to be assigned to a vector $(c_1(x), \ldots, c_k(x))$, where $c_i(x) \in [0, 1]$ denotes the degree (or proportion) in which object x belongs to class i, $i = 1, \ldots, k$. If the soft class estimations c_i actually represent proportions, it is then usual (see e.g. [9]) to impose the constraint

$$\sum_{i=1}^{k} c_i(x) = 1 \tag{1}$$

A soft classification problem can be addressed in either a supervised or unsupervised way, depending on the available data and the specific context requirements. For instance, in the complaint classification context sometimes an unsupervised approach may be more adequate, since the number k of possible complaint causes may not be perfectly determined *apriori*, and new complaint causes may always arise that are not present in the supervised data. At this respect, let us recall that soft supervised data is not necessarily needed to fit a soft supervised classification model. Even when trained with crisp data, most current supervised classification methodologies produce some kind of soft scores (probabilities, fuzzy degrees, etc.) in an intermediate stage of their process, before applying a decision rule (typically the well-known maximum rule) to map these soft scores into a crisp, single-class output (and see [10] for a discussion on potential drawbacks of such a decision rule).

In either way, whenever a classification problem is modeled in a soft (supervised or unsupervised) form, the crucial question is how the resulting model's performance is going to be evaluated. In this sense, it is important to notice that, if a practical task is modeled in soft terms, but crisp supervised data is used to evaluate the resulting soft model, then many misleading situations may occur.

To see it, suppose two different soft classifiers SC_1 and SC_2 are fitted to the same data in a binary classification task, in such a way that for a given object $x \in X$ the soft scores they respectively assign are $SC_1(x) = (0.51, 0.49)$ and $SC_2(x) = (0.49, 0.51)$. Suppose also that the crisp supervision of x has assessed it actually belongs to the first class, i.e. the correct crisp degrees are $(1, 0)$. Then, if the soft scores $SC_1(x), SC_2(x)$ are mapped into a single class through the maximum-rule, then the first classifier would predict x into the first class,

while the second classifier would predict it into the second one. Therefore, one classifier would be evaluated as correctly classifying x while the other would do it wrongly. However, the difference between the real soft outputs $SC_1(x)$ and $SC_2(x)$ is actually not as significant. Even more, suppose the actual soft class-composition of object x before the crisp supervision is $(0.6, 0.4)$. In this situation, both SC_1 and SC_2 would be doing a similar more or less accurate estimation of such a class mix, but imposing a crisp evaluation framework would lead to a totally different assessment of their performance. And if the soft degrees for x and $SC_2(x)$ remain as before, but now it is $SC_1(x) = (0.98, 0.02)$, then SC_1 would be committing a much greater error than SC_2, though however a crisp evaluation would just assess that SC_1 is right and SC_2 is wrong.

These reasons led us to devise a procedure to generate soft reference datasets for topic identification, that can enable the performance of soft classification procedures to be properly evaluated.

3 Soft Reference Documents Generation

This section is devoted to describe the proposed method to automatically generate soft reference documents for the task of topic identification.

Basically, the method departs from a dataset or corpus of documents, each of which is associated in a crisp way to a single topic, and generates an output corpus containing new documents in which the specified topics from the original documents are mixed following different randomly determined proportions.

More specifically, the inputs of the method are the following:

- *InputData*: The name of the input database containing the topic identification corpus. This database has to contain the following fields:
  - *documentID*: An identifier for each of the documents in the database. This identifier is a primary key and, as such, it can not present repeated values.
  - *text*: A free-format character field with the text associated to each document.
  - *topic*: A character variable identifying the topic associated to each document.
- *Topics*: This parameter specifies the topics that will take part in the generation of the documents with mixed topics in the output corpus. These topics names have to be contained in the set of topic names under the *topic* field of the database.
- *OutputCorpusSize*: This parameter specifies the size of the output corpus, that is, the number of documents it has to contain.
- *OutputData*: The name of the output corpus to be generated. This corpus will contain the following fields:
  - *documentID*: An identifier for each of the documents in the output corpus. As before, this identifier is a primary key and, as such, it will not present repeated values.
  - *text*: A free-format character field with the text of each document.

- *topicProportion*[*i*]: A numerical variable giving the proportion of phrases in each document that are associated to the *i*-th topic provided in the *Topics* parameter of the method. Therefore, the output corpus will contain as much *topicProportion* variables as different topics have been selected in the *Topics* parameter to be mixed in the output documents.

Regarding the method itself, it proceeds as follows:

1. For each topic i in the *Topics* parameter, a list L_i is generated that stores in each position a different phrase of the documents in the input corpus *InputData* associated to topic i. Therefore, the length of L_i is equal to the total number of phrases of the set of documents in the input corpus associated to topic i. We consider that a 'phrase' is any text between two consecutive periods, or the text between the beginning of a document and the first period.
2. Another list L is generated with as many positions as documents in the input corpus associated to any of the topics in the *Topics* parameter. Let us denote by N the length of this list. The number of phrases of each document in the input corpus associated to any of the topics in the *Topics* parameter is stored at the corresponding position of L. We shall use this list to create documents in the output corpus in such a way that their number of phrases follows a similar distribution to that of the documents in the input corpus.
3. For each $j = 1, \ldots, OutputCorpusSize$, a random number k between 1 and N is generated, and the number of phrases to be placed in the j-th document of the output corpus is assigned as $L[k]$. Then, draw $|Topics| - 1$ random numbers from a uniform $U(0, 1)$ distribution, and sort them in ascending order, so that $u_{(l)}$ denotes the l-th element of the sorted sequence. Assign $u_{(0)} = 0$ and $u_{(|Topics|)} = 1$. For $i = 1, \ldots, |Topics|$, select $(u_{(i)} - u_{(i-1)}) \cdot L[k]$ phrases at random from L_i, and successively write them in the *text* field of the j-th document of the output corpus. Similarly, for each document of the output corpus assign $documentID = j$ and $topicProportion[i] = u_{(i)} - u_{(i-1)}$. This completes the construction of the output corpus *OutputData*.

Figure 1 illustrates the generation of an *OutputData* document from the N documents of the *InputData* corpus that we assume deal with 3 selected topics. List L records the number of phrases of each of these N documents. Lists L_1, L_2, L_3 respectively contain the phrases of the documents associated to each of the 3 topics, and thus it is $N = |L_1| + |L_2| + |L_3|$. The number of phrases to be contained in document 1 of *OutputData* (let us refer to it as *OutDoc*1) is obtained by randomly selecting a value from L, say 8 (second position in L). Then, two random $U(0, 1)$ values are drawn and sorted, and stored as $u_{(1)}, u_{(2)}$. Let say we get $u_{(1)} = 0.125, u_{(2)} = 0.625$. Hence, 12.5% (0.125) of the 8 phrases in *OutDoc*1 are to come from topic 1, 50% (0.625–0.125) from topic 2, and 37.5% (1–0.625) from topic 3. Applying these proportions to the 8 phrases of *OutDoc*1, we get that 1,4 and 3 are the number of phrases to be respectively taken from topics 1 to 3. These number of phrases are then randomly drawn from lists L_1, L_2 and L_3, respectively, and written in *OutDoc*1. Notice that this draw is made with replacement, and thus some of the input phrases (as for instance

$Phrase_{2,|L_2|}$ in Fig. 1) may be repeated in the output documents. The reason behind this selection with replacement is that the number of phrases to select from a topic i may be greater than the number of phrases in the corresponding list L_i.

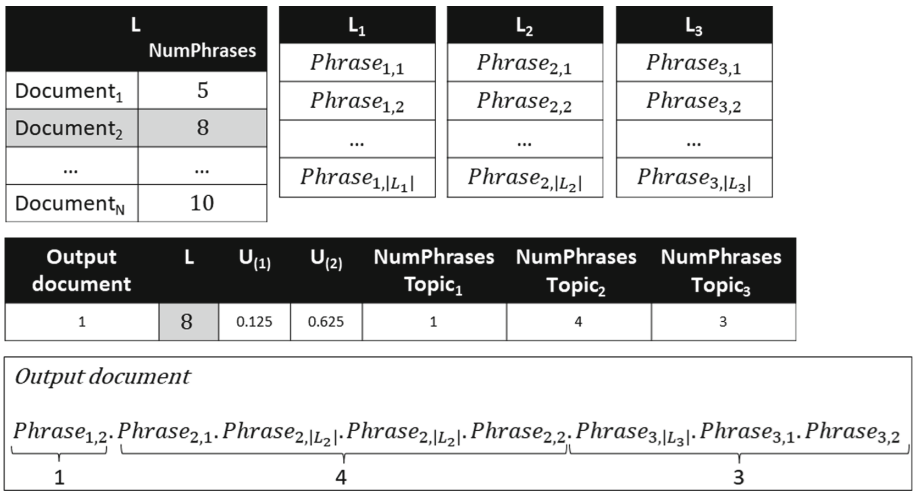

Fig. 1. Example of output document generation combining 3 topics.

Therefore, after this method is applied, a new corpus of documents is produced, in such a way that each of the new documents mixes text from the specified topics in different proportions. As these proportions are recorded together with the new text, the new corpus constitutes a soft reference dataset for topic identification. Furthermore, a main feature of the corpus provided by the proposed method is that the soft reference scores (i.e., the proportions of the different topics) assigned to each document are obtained in an objective way, not relying on subjective judgements from human supervisors.

4 Computational Study

In this section, we carry out an small computational experiment on unsupervised soft topic identification with data obtained by applying the soft reference generation method introduced in last section. The aim of this study is to illustrate the application of the proposed method on real data, as well as to provide a comparison of some well-known unsupervised classification techniques on soft reference data.

Therefore, this setting somehow mimics a real-world situation (as that described in Sect. 2 regarding soft complaint classification) in which a corpus of documents is available, in such a way that each document is known to simultaneously deal, up to a (possibly unknown) degree, with a set of topics. From this

knowledge, a model is searched that allows quantifying the proportion up to which each document deals with each of the topics being considered. To this aim, we contemplate the following steps:

1. Let k denote the number of topics being considered. Configure a soft reference corpus by applying the method described in Sect. 3, in such a way that each document i in this corpus mixes the k topics in proportions $p_i \in [0,1]^k$.
2. Apply natural language processing (NLP) techniques to translate the unstructured textual information of the documents in the corpus into a matrix with as many rows as documents, and with as many columns as relevant terms, so that it can be processed in the next step.
3. Apply an unsupervised soft classification algorithm to the matrix obtained in the previous step, only assuming that the number of topics k is known. The output of this algorithm then provides an estimation $\hat{p}_i \in [0,1]^k$ of the topics' weights for each document.
4. Obtain an estimation of the performance of the unsupervised algorithm in the task to be performed by comparing the estimations $\hat{p}_i$ with the real proportions p_i.

In the present study we focus on comparing the performance of two unsupervised techniques, the classic k-means (KM) algorithm and the fuzzy k-means (FKM). To this aim, we apply a fuzzyfication step from the final centroids provided by the KM algorithm, so that a soft output is obtained that can be compared in fair terms with that of the FKM. Taking into account that the documents to be clustered present a mix of topics, and thus that the classification variable to predict is soft in nature, the hypothesis we would like to test is whether the FKM outperforms the post-fuzzified KM in providing a more accurate estimation of the actual weights of the topics in the documents to be processed.

4.1 Experimental Setting

Let us now describe the details of the computational study carried out. First of all, we applied the proposed soft reference generation method on the 20-Newsgroup (NG20) dataset [11], which constitutes a well-known benchmark in topic identification. This corpus contains a total of 20,000 documents, with exactly 1000 texts dealing with each of the 20 different topics shown in Table 1.

The proposed method has been setup to provide two different corpus typologies:

- A first kind of corpus contains documents mixing the topics of Atheism (NG1) and Graphics (NG2). Therefore, in this case it is $Topics = \{\text{NG1 NG2}\}$, and both topics are combined in proportions $p_i = (u_i, 1-u_i)$, where u_i is a random $U(0,1)$ value drawn for each document i.
- A second kind combines the topics of Graphics (NG2), Baseball (NG10), and Space (NG15). Then, now it is $Topics = \{\text{NG2 NG10 NG15}\}$, and for each document i a pair of random $U(0,1)$ values u_i, v_i are drawn. Assuming without loss of generality that $u_i < v_i$, the proportions of the mix of the 3 topics are given by $p_i = (u_i, v_i - u_i, 1 - v_i)$.

Table 1. Thematic blocks and topics of the 20-Newsgroup dataset.

Block	Topics [NG]
Alternative	Atheism [1]
Computing	Graphics [2], os.ms-windows [3], sys.ibm [4], sys.mac [5], windows.x [6]
Miscellaneous	Forsale [7]
Recreation	Autos [8], motorcycles [9], baseball [10], hockey [11]
Science	Cryptography [12], electronics [13], medicine [14], space [15]
Social issues	Religion.christian [16]
Talk	Guns [17], Mideast [18], politics.miscellaneous [19], religion.misc [20]

Each of these corpus typologies will be used in a different comparison experiment. To provide a more robust comparison through non-parametric statistical tests, 30 corpus of each typology are produced, each containing $OutputCorpusSize = 1000$ documents.

Once these 60 corpuses were generated through the proposed method, the following NLP steps were applied to each of the 60,000 documents:

- Tokenization [12]: The text of each document is separated in tokens (terms), generating a variable associated to each token.
- Stopwords removal: Non-significant words as conjunctions, determiers, etc. are removed.
- Stemming [13]: Process by which each token is reduced to its root form, so that different inflections are concentrated in a single token independent of number, gender, verbal conjugation, etc. To this aim, the Porter algorithm [14] has been applied.

As a result, for each corpus a matrix with as many rows as documents and as many columns as tokens is produced. Each position of this matrix is filled with the $tf - idf$ metric [15], that represents the relative frequency (term frequency, tf) of each token in a document, penalizing those words that appear with a relatively high frequency in a corpus. This penalization (applied through the so called inverse document frequency, idf) is introduced since a given term should not characterize too strongly a document whenever that term appears frequently in other documents of the corpus. Specifically, the $tf - idf$ metric is defined as follows:

$$tf - idf_{ij} = tf_{ij} \cdot idf_j, \tag{2}$$

where

$$tf_{ij} = \frac{\#\ of\ times\ term\ j\ appears\ in\ document\ i}{total\ \#\ of\ terms\ in\ document\ i} \tag{3}$$

and

$$idf_j = log\left(\frac{total\ \#\ of\ documents}{1 + total\ \#\ of\ documents\ containing\ the\ term\ j}\right). \tag{4}$$

Once these $tf - idf$ matrices are produced, the textual information gets structured in a form allowing the application of unsupervised classification techniques. However, before that, some dimension reduction steps are applied. Firstly, as the corpuses still contain a huge number of tokens (variables), some of which may occur quite infrequently within a corpus, a relevance thresholding step is applied. This consist in removing those tokens for which their accumulated $tf - idf$ in a corpus is lower than a given threshold. This threshold, known as the transition point [16], is set to the 95th percentile of the distribution of accumulated $tf - idf$ of all tokens in the corpus.

Finally, principal component analysis (PCA) has been applied on the remaining tokens in order to further reduce the dimension of the corpus matrices to a few variables. The number of components to use for each corpus typology has been determined through sedimentation graphs. As shown in Fig. 2, the number of components to be used depends on the number of topics considered. Particularly, just the first two components are retained for corpuses of the first typology (NG1/NG2), and three for corpuses of the second typology (NG2/NG10/NG15). Therefore, a 1000×2 matrix is finally obtained for each corpus of the first type, and a 1000×3 matrix is associated to each corpus of the second type.

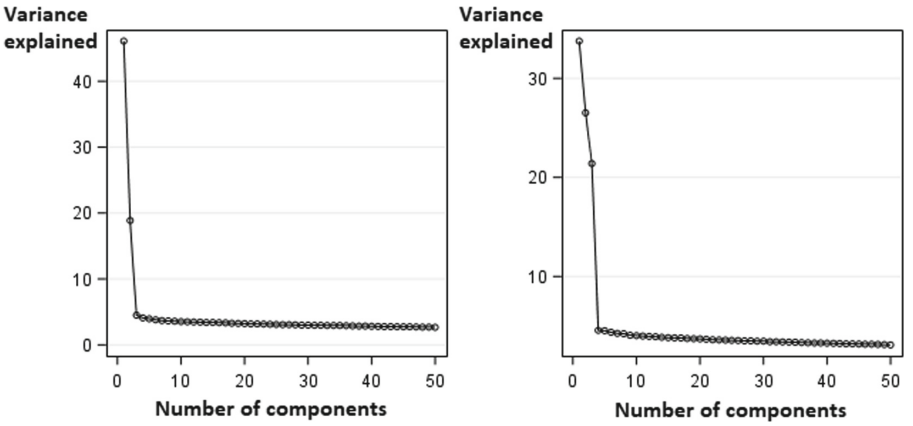

Fig. 2. Sedimentation graphics NG1/NG2 (left) and NG2/NG10/NG15 (right).

The KM and FKM algorithms are then applied on these matrices. The number of clusters k to form are set to the number of topics being combined for each corpus, i.e. $k = 2$ for the first typology and $k = 3$ for the second one. In both methods, 30 random starting centroids are tried for each corpus, allowing a maximum of 100 iterations after each initialization. Only the best result in terms of the intra-cluster variance objective function is keep. In all runs of the FKM the fuzziness parameter was set to $m = 2$.

As KM actually provides a crisp output, but the reference class variable is soft, a fuzzification step is needed in order to allow a fair comparison. This step is

applied only after the best final KM centroids (out of the 30 random starts) are obtained. Specifically, the applied post-fuzzification follows the same scheme used in the FKM to obtain degrees from centroids in each iteration. Particularly, as in the FKM, the fuzzified degrees sum up to 1, and thus they can be interpreted as proportions. Therefore, let d_{ij} denote the Euclidian distance between document i and the j-th centroid ($j = 1, .., k$) reached by KM. Then, for the sake of the comparison with the soft reference, we consider the KM estimation of the proportion in which document i deals with topic j to be given by

$$\hat{p}_{ij} = \left(\sum_{l=1}^{k} \frac{d_{ij}^2}{d_{il}^2} \right)^{-1} \tag{5}$$

The performance metric, actually an error measure, will be given by the mean difference between the proportions estimated by the clustering methods $\hat{p}_i = (\hat{p}_{i1}, \ldots, \hat{p}_{ik})$ and the real proportions $p_i = (p_{i1}, \ldots, p_{ik})$ [17]. Thus, for a corpus with S documents and k topics, the error measure is given by

$$err_{Corp} = \frac{1}{Sk} \sum_{i=1}^{S} \sum_{j=1}^{k} |p_{ij} - \hat{p}_{ij}| \tag{6}$$

Finally, let us mention that a matching step has to be applied due to the unsupervised character of the algorithms considered. A clustering method estimate k proportions, but the order in which they appear does not have to be the same as that of the reference clusters. This matching process is applied by corpus.

4.2 Results

This section presents the results of the two experiments carried out to compare the performance of the (post-fuzzified) KM and the FKM algorithms on the described soft topic identification task. As exposed above, both comparisons are performed on a set of 30 corpus generated by the proposed method. Each corpus contains $S = 1000$ documents, that combine $k = 2$ topics in the first comparison and $k = 3$ topics in the second one.

The SAS software has been used to implement the soft reference generation method, as well as for preprocessing the data and perform PCA. The R software (packages *stats* and *fclust*) has been used for fitting the models and compute the errors shown in this section.

Table 2 shows the mean error of the KM and FKM algorithms and its standard deviation for each of the 30 corpuses analyzed. Clearly, in average both algorithms estimate the real proportions of each topic with similar accuracy. However, it is also important to notice that FKM tends to consistently produce slightly lower error rates than KM. Following [18], to rigorously analyze the statistical significance of this behaviour, a Wilcoxon signed rank test is applied on the results in Table 2. The results of this test are shown in Table 3, from which it is possible to conclude that FKM provides corpus error rates with a significantly lower median than KM.

Table 2. Error by corpus (mean ± standard deviation)

	2 Topics		3 Topics	
	FKM	KM	FKM	KM
1	.146 ± .110	.146 ± .109	.120 ± .069	.135 ± .081
2	.147 ± .107	.149 ± .109	.123 ± .070	.127 ± .072
3	.135 ± .100	.138 ± .103	.121 ± .070	.127 ± .075
4	.135 ± .103	.137 ± .104	.130 ± .072	.142 ± .079
5	.129 ± .100	.129 ± .100	.119 ± .068	.120 ± .070
6	.145 ± .106	.147 ± .107	.125 ± .068	.133 ± .075
7	.144 ± .109	.148 ± .112	.123 ± .073	.132 ± .077
8	.153 ± .114	.156 ± .117	.123 ± .071	.121 ± .070
9	.138 ± .103	.138 ± .103	.116 ± .070	.120 ± .070
10	.139 ± .101	.140 ± .103	.117 ± .067	.117 ± .067
11	.140 ± .104	.141 ± .106	.127 ± .074	.133 ± .079
12	.144 ± .111	.144 ± .111	.137 ± .075	.144 ± .079
13	.135 ± .110	.135 ± .109	.122 ± .072	.125 ± .074
14	.128 ± .103	.127 ± .102	.115 ± .070	.113 ± .068
15	.138 ± .104	.140 ± .106	.124 ± .073	.130 ± .077
16	.146 ± .102	.145 ± .102	.119 ± .071	.122 ± .072
17	.146 ± .112	.146 ± .114	.114 ± .069	.122 ± .071
18	.130 ± .098	.132 ± .098	.119 ± .070	.133 ± .080
19	.147 ± .108	.149 ± .111	.123 ± .073	.135 ± .076
20	.140 ± .107	.143 ± .109	.112 ± .066	.124 ± .075
21	.140 ± .106	.140 ± .106	.124 ± .076	.133 ± .078
22	.142 ± .109	.144 ± .111	.121 ± .068	.124 ± .071
23	.136 ± .107	.137 ± .109	.119 ± .076	.123 ± .077
24	.140 ± .101	.141 ± .102	.112 ± .065	.112 ± .064
25	.141 ± .101	.144 ± .104	.115 ± .068	.116 ± .067
26	.141 ± .107	.140 ± .106	.123 ± .069	.210 ± .113
27	.141 ± .103	.144 ± .104	.123 ± .073	.199 ± .106
28	.151 ± .110	.152 ± .111	.125 ± .076	.129 ± .079
29	.146 ± .106	.145 ± .105	.119 ± .070	.125 ± .072
30	.153 ± .106	.159 ± .112	.124 ± .071	.124 ± .072
Mean	**.141 ± .0064**	.143 ± .0071	**.121 ± .0063**	.132 ± .021

In summary, the final centroids produced by the KM and FKM algorithms provide a similar degree of accuracy when estimating the topic composition of the documents in the generated soft reference corpuses. Nevertheless, the inner fuzzification step of the FKM seems to slightly but consistently improve the final

Table 3. Wilcoxon test to compare fuzzy k-means (R^-) against k-means (R^+). For each experiment the differences in error rates for each corpus i are expressed as $FKM_i - KM_i = sign(FKM_i - KM_i)\,|FKM_i - KM_i|$, and sorted in increasing order of the absolute differences. This allows assigning a rank to each difference. The sums of ranks of positive and negative difference are denoted by R^+ and R^-. Under the null hypothesis of equal median error rates these statistics should be similar.

Comparison		R^-	R^+	p-val
FKM vs. KM	2 Topics	431	34	**6.918e-06**
FKM vs. KM	3 Topics	451	14	**2.049e-07**

FKM centroids with respect to those finally achieved by the KM algorithm, which instead performs a crisp cluster assignment in each iteration and has only been fuzzified after its conclusion.

5 Conclusions

A method to generate documents to be used as soft reference data in topic identification tasks has been introduced in this work. The method proceeds by combining phrases of previously available texts associated to different topics, in random but known proportions. As a consequence, a main feature of this method is that it allows generating reference data with objective soft degrees, not relying on subjective judgements from human supervisors. Reference corpuses containing any number of documents that combine a wide variety of topics can be thus obtained through the proposed method. Corpuses of this kind can then be used to allow evaluating the performance of soft topic classification techniques, both supervised and unsupervised. We consider this a relevant contribution as soft reference data for topic identification were nonexistent or hardly available.

A complete computational study was also carried out in this work, illustrating the application of the proposed method on real data and showing the possibility of conducting a proper comparison of the performance of two soft unsupervised classification methods. Particularly, this study allowed to conclude that the inner fuzzification step of the fuzzy k-means algorithm provide slightly but consistently better centroids for soft topic identification than those of the classic k-means algorithm, at least under certain conditions of the soft reference documents.

Future work regarding the proposed method will consider its extension to allow generating documents that combine phrases of different subsets of the specified topics, not necessarily combining all them simultaneously. This will allow more realistic soft reference documents to be generated, particularly for the context of complaint classification. Further work will also be devoted to assess the performance of soft unsupervised methods under various probability distributions of the random numbers used to determine the proportions of phrases of the different topics being combined.

References

1. Chakraborty, G., Pagolu, M., Garla, S.: Text Mining and Analysis. Practical Methods Examples and Case Studies Using SAS, p. 1. SAS Institute Inc., Cary (2013)
2. Berry, M.W., Koban, J.: Text Mining: Applications and Theory. Wiley, Hoboken (2010)
3. Duda, R.O., Hart, P.E.: Pattern Classification and Scene Analysis. Wiley, New York (1973)
4. Swam, P.H., Davis, S.M.: Remote Sensing: The Quantitative Approach. McGraw-Hill, New York (1978)
5. Settle, J.J., Drake, N.A.: Linear mixing and the estimation of ground cover proportions. Int. J. Remote Sens. **14**(6), 1159–1177 (1993)
6. del Amo, A., Montero, J., Fernandez, A., Lopez, M., Tordesillas, J.M., Biging, G.: Spectral fuzzy classification: an application. IEEE Trans. Syst. Man Cybern. Part C **32**(1), 42–48 (2002)
7. Binaghi, E., Brivio, P.A., Ghezzi, P., Rampini, A., Zilioli, E.: Investigating the behaviour of neural and fuzzy-statistical classifiers in sub-pixel land cover estimations. Can. J. Remote Sens. **25**(2), 171–188 (1999)
8. Wang, S., Wu, B., Wang, B., Tong, X.: Complaint classification using hybrid-attention GRU neural network. In: Yang, Q., Zhou, Z.-H., Gong, Z., Zhang, M.-L., Huang, S.-J. (eds.) PAKDD 2019. LNCS (LNAI), vol. 11439, pp. 251–262. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-16148-4_20
9. Ruspini, E.H.: A new approach to clustering. Inf. Control **15**, 22–32 (1969)
10. Villarino, G., Gómez, D., Rodríguez, J.T., Montero, J.: A bipolar knowledge representation model to improve supervised fuzzy classification algorithms. Soft. Comput. **22**(15), 5121–5146 (2018). https://doi.org/10.1007/s00500-018-3320-9
11. Hettich S., Bay S.D.: The UCI KDD archive, pp. 1721–1288. University of California, Department of Information and Computer Science, Irvine, CA (1999). http://kdd.ics.uci.edu
12. Hassler, M., Fliedl, G.: Text preparation through extended tokenization. University Klagenfurt (2006)
13. Jivani, A.G.: A comparative study of stemming algorithms. Department of Computer Science and Engineering, the Maharaja Sayajirao University of Baroda Vadodara, Gujarat, India (2011)
14. Porter, M.F.: An Algorithm For Suffix Stripping, Readings in Information Retrieval, pp. 313–316. Morgan Kaufmann Publishers, Inc., Burlington (1997)
15. Salton, G., Yang, C.S.: On the specification of term values in automatic indexing. J. Doc. **28**(1), 11–21 (1973)
16. Pinto, D., Rosso, P., Jimenez-Salazar, H.: A self-enriching methodology for clustering narrow domain short texts. Comput. J. **54**, 1148–1165 (2011)
17. Gómez, D., Biging, G., Montero, J.: Accuracy statistics for judging soft classification. Int. J. Remote Sens. **29**(3), 693–709 (2008)
18. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power. Inf. Sci. **180**(10), 2044–2064 (2010)