A New Two-Stage Approach to the Multiaspect Text Categorization

Sławomir Zadrożny, Janusz Kacprzyk, IEEE Fellow, and Marek Gajewski
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6, 01-447 Warszawa, Poland
Email: {Slawomir.Zadrozny,Janusz.Kacprzyk}@ibspan.waw.pl

***Abstract*—We consider a particular type of text categorization problem which we refer to as the multiaspect classification. It is inspired by some practical scenario of business documents management in a company but has a broader application potential. A distinguishing feature of the new problem considered is the existence of two schemes of classification. The first one is based on the traditional, static set of text categories, possibly arranged into a hierarchy. The second one is based on a dynamic structure of sequences of documents, referred to as *cases*, identified within each category. While the former problem may be addressed using one of the well known techniques of text categorization (classification), the latter seems to require some distinct approaches due to the fact that the set of cases is unknown in advance, as well as due to the assumed limited number of training documents, if a case should be interpreted as a classic category. In the paper, we discuss the problem in a more detail as well as show the applicability of an intuitively appealing two stage approach to solving the problem of such a multiaspect text categorization.**

I. Introduction

Textual information retrieval (IR) is one of the traditional branches of computer science which can be traced back to the middle of the previous century, and in fact much more back into the past if meant as a set of techniques to effectively and efficiently manage the information that is in real world textual in a considerable part . Nowadays, IR is still in the center of interest of the research community and there seem to be two main driving forces behind that. First of all, an enormous amount of textual information have been gathered in an electronic form due to a widespread use of computer-based information systems. Second, the gathering of this information and the access to it has been made much easier thanks to the popularity of the Internet and its related information processing technologies. Thus, there is a need to develop new techniques to deal with a rapidly growing size of documents collections as well as with new tasks related to the textual information processing.

The classic task of IR consists in retrieving documents which are relevant with respect to user information needs formulated as a query against a collection of documents. There are however many more related tasks, notably text categorization [1]. Its most useful form, from a practical point of view, is dealt with in the literature using one of many well-known supervised learning techniques. In our previous papers [2], [3] we introduced a basic idea of a new problem of text categorization which we propose to call *multiaspect text categorization*. It is inspired by a practical problem of document management in a company and its distinguishing feature is that two categorization schemes are taken into account.

That idea of a new problem formulation of multiaspect text categorization has then been further extended in our next papers, and is also dealt with here, notably by pursuing a new line of reasoning the essence of which is that a two stage problem solution structure each involving relatively simple classifiers can yield a good solution, and be effective and efficient. One part/stage of this problem is a more or less standard text categorization against a hierarchy of categories, cf., e.g., [4]–[6]. The second stage consists in assigning a document to a sequence of documents referred to as a *case*. The following assumptions are important to fully specify the whole problem: all documents of a case belong to the same category, and documents are to be classified one by one and are available for classification in an order which basically reflects the chronology of their creation (appearance).

The very specifics of the problem under consideration is mainly related to the second stage. Formally, the cases may be treated as additional, low-level categories. However, they are not known in advance, as opposed to "regular" categories that are usually assumed to be so. Actually, a new document may be classified/assigned to an existing case but may also start a new case. Hence, here we go well beyond the standard text categorization problem where the categories are predefined. Moreover, the cases are assumed to be relatively small, which is usually the case in practice, and this additionally implies difficulties for usual approaches to supervised learning. It should be expected that a key to the successful assignment of documents to appropriate cases is to have means to effectively represent the order (succession) of documents in cases belonging to the same category. If the existence of such a pattern may be assumed, then this should greatly help to decide which case a given document belongs to. Such an assumption is well-founded for the practical scenarios which were the inspiration for our work. For example, to be more specific, let a given category comprise documents related to tenders carried out by a company to purchase some materials needed. Then, the cases in such a category consist of chronologically

978-1-4799-7560-0/15 $31.00 © 2015 IEEE
DOI 10.1109/SSCI.2015.210

ordered documents produced and received during the execution of a tender. Therefore, one may expect that the succession of documents related to different tenders follows a pattern like that: first there is an internal order for the materials which triggers the announcement of a tender, then offers are received, etc.

Our work on the multiaspect text categorization goes in two directions. First, we look for ways to model, discover and exploit, for the purposes of classification, the above mentioned patterns in sequences of documents. To this aim, in [2], [7] we have applied the sequence mining techniques (cf., e.g., [8]) as well as we have proposed to use the Hidden Markov Models (HMMs). We have also looked for a possibility to apply other techniques from different fields, such as data mining methods [9]–[12] or methods with roots in musical information processing [13]. Second, we have been looking for the solution to the problem stated using some more traditional approaches based, in general, on the concept of matching between two or more documents, without resorting to, e.g., an explicit modeling of patterns present in sequences of documents. In [3] we have proposed to employ the concept of the fuzzy subsethood to measure the similarity of documents. In fact, devising such a measure, to be appropriate for the task at hand, seems to be the most important component of any effective solution technique of the problem. Some of the work on data mining techniques mentioned earlier in the context of the first line of research is applicable also here. We have also looked for some other techniques that may be applicable known in other related fields such as data quality maintenance [14].

The current paper belongs to the second line of research as mentioned above. We propose a two stage categorization approach composed of a standard k-NN technique for a category selection and for a case selection a variant of k-NN taking into account the sequential nature of cases.

The organization of the paper is the following. First, we formally define the multiaspect text categorization problem and briefly discuss approaches to similar problems known from the literature as well our own solutions proposed earlier. Then, we propose a new solution approach and report some preliminary results of the computational experiments. We conclude with a summary of the results obtained and plans for further research.

Briefly speaking, the paper presents first in a comprehensive way a novel multiaspect text categorization problem, some techniques that were proposed earlier to solve it, with those employed for solving closely related problems, and, finally a proposal of a new approach and results of some computational experiments.

II. Multiaspect Text Categorization and Related Work

A. Problem Formulation

We define the problem of the multiaspect text categorization from the perspective of an computer-based information system acting according to the paradigm of the computer assisted/supported decision making. We assume a collection of documents

$$D = \{d_1, \ldots, d_n\} \tag{1}$$

has been gathered which are managed by a company as mentioned in the Introduction. Thus, the documents are classified to a number of predefined *categories* or a hierarchy of such categories. At this level, using the standard terminology, a multiclass, single label classification problem may be recognized: there is a set of predefined categories, denoted as

$$C = \{c_1, \ldots, c_m\} \tag{2}$$

and each document $d \in D$ belongs to exactly one class $c \in C$.

The documents are however further structured within particular categories. Namely, each document d belongs additionally to a *case*, being a sequence of documents, and each document d belongs to exactly one sequence.

The particular cases will be denoted as σ and their set as Σ:

$$\sigma_k =< d_{k_1}, \ldots, d_{k_l} > \tag{3}$$

$$\Sigma = \{\sigma_1, \ldots, \sigma_p\} \tag{4}$$

To summarize, each document $d \in D$ belongs to one category c_j and to one case σ_j, and all documents from the same case belong to the same category.

Now, a new document d^* appears and has to be added to the collection D and the system is meant to support a human user responsible for its classification to a category $c \in C$ and to a case $\sigma \in \Sigma$. One can try to assign d^* directly to a case σ as the assignment to a case implies also a category c in which the case σ is located. However, this is a difficult problem as cases treated as categories will be usually represented by a small number of documents and their set is not prespecified. A new document may be related to a new case which was not seen before. Due to the difficulties indicated, it may be worthwhile to first assign the document to a category and only then choose a proper case within it. In fact, both classifications may be combined and support each other as it is the case of our approach proposed in [3]. As the set of categories is prespecified and each of them may be assumed to be represented by a sufficient number of documents in the collection D, then the standard text categorization techniques may be employed; cf., e.g., [1]. In practical scenarios these assumptions will be usually satisfied. In case of a company managing a collection of documents, which is our primary inspiration of a possible application of the proposed approach, the documents are classified on a yearly basis but for the training data set there may be employed all the documents belonging to a given category and collected in the previous years.

It is worth to analyze the MTC problem from the point of view of the semi-supervised learning paradigm. In the general formulation of the problem given above, this aspect seems to be relevant. The newly arriving documents which are automatically classified by the system may play their role in the classification of the further documents. In the general

text categorization problem they may be ignored, i.e., only the initial set of documents may be a basis for the classification of any future documents. In the MTC, taking into account the essence of the problem these documents should definitely be considered for the classification of further documents. We assume, as mentioned earlier, that there is some logical succession of documents within the cases and thus the newly coming documents may be crucial for the classification of the further documents, arriving after them. Then, there is a question if the documents classified by the system should be treated during the further classification as unlabelled, i.e., their usage would be possible only in the spirit of the semi-supervised learning, or should be assumed to be properly classified and later on used in the same way as other documents of the initial training data set. We definitely follow the second alternative as it is more in line with the practical scenarios being the motivation for our formulating and considering the multiaspect categorization problem. Namely, a document arriving at the institution has to be classified at once and the decision on that is practically final, irrevocable. Of course, under such an assumption the role of the classification system should be seen more in the vein of the decision making aid paradigm than the full automating of the decision regrading proper classification of a document.

Basically, the problem of the multiaspect text categorization (MTC), briefly sketched above, may be considered as a two level text categorization problem. It has been introduced by us in [2], [7]. The most similar problem known in the literature is that of the Topic Detection and Tracking (TDT) [15]. TDT was a part of the DARPA Translingual Information Detection, Extraction, and Summarization (TIDES) program, closely related to the well-known Text REtrieval Conferences (TREC). Research on the TDT started in 1997 [16] and was followed by regular workshops during the next seven years. Topic detection and tracking is considered in the framework of processing of a stream of news which are coming from various sources and bring information on a set of events. The basic task is here to group together news stories describing various aspects of the same events. In order to reveal an intrinsic relation to our multiaspect text categorization problem let us first align the terminologies used here and there. An individual news in TDT is referred to as a *story* and it corresponds to a document in our MTC. Stories in TDT describe *events* and some major events together with interrelated minor events are referred to as *topics* and correspond to both categories and cases in MTC with an emphasis on the latter. Topics, similarly to cases, are not predefined and new topics have to be *detected* in the stream of stories and then *tracked*, i.e., all subsequent stories dealing with the same major event have to be recognized and classified to a topic detected earlier.

There is a number of task distinguished within TDT. From our perspective the most important are *topic detection* and *first story detection*. The former may be identified with the classification of documents to the cases in our MTC: starting with a (possibly empty) set of groups of stories forming particular topics, a new incoming document has to be assigned to one of these topics or to form a new topic. The latter task is, in fact, a part of the former and consists in recognizing if a document belongs to one of the earlier detected topics or is the first story of a new topic.

The main differences between TDT and MTC may be briefly stated as:

1) categories and cases are considered in MTC as opposed to topics only in TDT,
2) cases are sequences of documents while topics are basically just sets of stories; even if stories are timestamped, their possible temporal type relations are not analyzed and the timestamps are only used to discount the information related to older stories,
3) there is a different practical inspiration for TDT and MTC which implies further differences in assumptions adopted in both cases (besides the two aspects mentioned above).

For a further analysis of relations of the multiaspect categorization problem and the topic detection and tracking problem the reader is referred to our forthcoming paper [17].

B. Earlier Solutions

Due to some similarity of MTC to TDT, some methods proposed for the latter may also be applicable for the former. For the purposes of TDT, the stories are most often represented using the vector space model [18] and are subject to classic text operations such as stopword elimination and stemming. The basic operation which is performed to accomplish particular TDT tasks is that of computing the similarity/dissimilarity of stories represented as vectors in a multidimensional space. Again, classic techniques of IR are employed such as the cosine of the angle between two vectors. The topic detection task then boils down to the incremental clustering task. A simple approach [16] consists in representing clusters via their centroids and comparing a new story to be classified with each centroid. The story is assigned to the cluster of which the centroid is most similar to the story. However, some threshold value is assumed and if this maximal similarity is not higher than that value, then a new cluster is formed and the story is assigned to it. Thus, the efficiency of solution of the first story detection problem boils down to adopting a proper threshold value.

In [3] we have proposed another solution approach based on the vector space model and a similarity measure to compare documents and their sets. We assume that the documents are rather short and thus the document-term matrix may be extremely sparse. Thus, we propose to employ an aggressive dimension reduction and to leave in the representation of a document only K keywords with the highest $tf \times IDF$ weight [18]. In our experiments reported in [3] the parameter K was set to 5. Then, each category was represented by a centroid, a vector being the mean of vectors representing training documents belonging to the given category. On the other hand, each case was represented by the vector formed as the union of keywords representing documents present in a given case. Finally, all vectors, representing: a document

to classify, categories centroids and cases, were treated as fuzzy sets in the space of keywords. The matching of a document and a case was computed as the weighted average of fuzzy subsethood degrees of the fuzzy set representing the document to a fuzzy set representing a given case and fuzzy set representing the category of this case, respectively. This way, the assignment of a document to a case for which the highest matching was obtained, which implied also the assignment to its category, was based on the combination of the matching of this document with respect to the case as well as to the whole category.

To further develop the above source approach, in [2] we proposed solutions based on an attempt to detect a pattern in cases belonging to the same category. First, the hidden Markov models (HMM) were employed to model such a pattern. For each category, a separate HMM was to be constructed to model its cases using standard methods based on training data [19]. The classification of a new document consisted in adding it to each candidate (on-going) case and checking for which case the document was the most probable extension.

Another approach proposed by us [2] was based on the sequence mining algorithm by Zaki [8]. Now, the patterns sought were sequences of keywords frequently appearing in the cases of a given category. A document to be classified was assigned to such a case for which it was the best extension in terms of matching the earlier discovered patterns.

The new solution approaches mentioned above proved to be effective and efficient, but – by leaving some analytic and algorithmic space for possible improvements – could be adopted as conceptual bases for extensions. In this paper we discuss another approach belonging to the first family of methods proposed earlier, i.e., based on a standard classification algorithm but addressing the specificity of the multiaspect text categorization.

III. A Two-Stage Approach

In the proposed approach we adopt the representation of documents based on the vector space model. The *tf* $\times$ *IDF* weighting scheme [20] is employed in the experiments reported in the next section. No special representation for the categories and cases is assumed as the classification is carried out using the k nearest neighbors algorithm (k-NN). The collection of documents D (1) is assumed to be structured as described earlier, i.e., each document belongs to a category and within a category to a case. Moreover, there are cases which are completed (closed) and other which are still open (on-going).

The classification of a new document d^* is carried out in two stages. First, the k-NN classifier is used to assign a category c^* to the document d^*. Thus, all documents of D are used as training documents and the k closest ones, in the sense of some distance measure, are used to decide the category of d^* via a usual majority. In the experiments we used the Euclidean distance and $k = 10$. Second, the last documents of all on-going cases belonging to the category c^* chosen at the first stage are compared to the document d^* and the case σ^* is assigned to d^* for which its last document is the closest one to d^*, in the sense of some distance measure assumed. Formally, this two-stage algorithm may be described as follows using the notation of (1)-(4):

$$c^* = \arg\max_{c_i} |\{d \in D : \text{Category}(d) = c_i \wedge d \in \mathbf{NN}_k(d^*)\} \tag{5}$$

where $\text{Category}(d)$ denotes the category $c \in C$ assigned to the document d and $\mathbf{NN}_k(d^*)$ denotes the set of k documents $d \in D$ closest to d^*. Then,

$$\sigma^* = \arg \min_{\sigma_{j, j \in O_{c^*} = \{1, \ldots, o_{c^*}\}}} \text{dist}(d^*, \mathbf{last}(\sigma_j)) \tag{6}$$

where O_{c^*} is the set of indices of all on-going cases belonging to the category c^* and $\mathbf{last}(\sigma)$ denotes the last document of the case σ.

The classification of a document to the case depends on the proper category assignment and on the assumption that any pair of neighboring documents in a sequence should be similar. This assumption may be not valid in general but is acceptable if a sequence of documents may be interpreted as having its roots in one long document which is splitted/segmented into several parts. We test our approach in the next section exactly on such a data set.

Our main goal is to check if such an inexpensive procedure of assigning documents to cases may work satisfactorily enough. It is computationally inexpensive, with the cost proportional to the number of on-going cases which may be assumed in practical scenarios quite limited.

The first story detection is solved within the framework of this approach using a threshold value. Namely, if the minimum value in (6) is below the threshold value, then a new case is formed within the category c^* with the document d^* as the starting document. Obviously, the threshold value should be selected experimentally. This approach requires further studying as the results of the experiments, not reported here, show that densities of distances between the first stories and the rest other documents are not that clearly separated from the distances between "non first story" documents.

IV. Experiments

A. The Documents Collection

Since the problem of the multiaspect text categorization as proposed in our works and described briefly here is a new one, then there are no available benchmark document collections to carry out numerical experiments. A possible option is to use collections used for some known topic detection and tracking problem related experiments. However, these collections are not perfectly fitted for our purposes due to inherent differences between the MTC and TDT problems as mentioned earlier. Anyway, we plan to adapt one of the TDT collections in our future work after a necessary adaptation. For the time being, in our work we are using the same document collection as used in [2], [3], i.e., the ACL ARC.

The ACL Anthology Reference Corpus (ACL ARC) [21] consists of selected scientific papers on computational linguistics. Each paper comprises a number of explicitly distinguished

sections. For our purposes, each paper yields a case σ and sections of this paper become documents of this case with their order (sequence) preserved. Originally, each paper is represented as an XML file. We process all XML files and produce for each a separate directory which contains text files representing particular sections of the paper. The names of the text files contain the number of a given section within a particular paper. We are using a subset of the ACL ARC comprising 113 papers and thus we get 113 cases with ca. 13 documents each, on the average. The papers in ACL ARC are not explicitly organized in categories. Thus, we employ the clustering algorithm to group the whole papers, as described in detail later on. This way we obtain 7 categories with ca. 200 documents each, on the average. Notice, that we are clustering the whole 113 papers and then each of them yields ca. 13 documents.

Then, we divide the whole collection into the training and testing data sets. To this aim, we randomly choose 20% of cases as on-going cases. The remaining cases form the first part of the training data set. In each on-going case we randomly select a cut-off point/index. All documents in the on-going cases which appear in their cases before the cut-off point are added to the training data set as its second part while the remaining documents, i.e., appearing at the cut-off point and later, form the testing data set. The whole training data set is used by the k-NN classifier to assign the category to the testing documents while only the last documents of the on-going cases are used to decide to which case a new document should be assigned.

Random selection of the testing documents and the cut-off points is performed using the R's `sample` function, i.e., via the uniform distribution, without replacement.

We implemented the algorithm on the R platform [22] using several packages, notably the `tm` package [23] to process the collection of text documents. The code preparing the data collection and implementing the categorization algorithm takes the form of a few R scripts. The document collection is prepared in the following steps (based on the set of text files representing the sections of the selected papers of the ACL ARC and arranged in directories corresponding to particular papers, as mentioned earlier):

1) creation of the corpus of all selected papers using the `tm::Corpus` function; for this purpose the whole papers are reconstructed from text files representing their sections;
2) the corpus created is normalized, i.e., punctuation, numbers and multiple white spaces are removed, stemming is applied, the case is changed to the lower case, stopwords and words shorter than 3 characters are eliminated;
3) a document-term matrix is constructed for the normalized corpus using the `tm::DocumentTermMatrix` function and *tf* $\times$ *IDF* terms weighting scheme; sparse keywords, i.e. appearing in less than 10% of documents are removed from the document term-matrix; finally, the vectors representing documents are normalized in such a way that each coordinate is divided by the norm of the vector;
4) the documents, i.e., the whole papers of the ACL ARC collection, are then clustered using the k-means algorithm implemented in R via the `stats::kmeans` function; this way 7 clusters are obtained which define the categories of particular documents in our collection; in [2], [3] in similar steps preparing the collection we remove clusters smaller than 10 documents but for the method proposed in this paper due to its use of the k-NN algorithm the number of cases per category seems to be less critical;
5) another corpus is created, this time comprising all documents, i.e., sections of the papers of the ACL ARC collection; it is normalized as previously, i.e., via the punctuation, numbers and multiple white spaces removal, stemming, changing all characters to the lower case, stopwords and words shorter than 3 characters elimination;
6) a document-term matrix is constructed for the above corpus corpus using the `tm::DocumentTermMatrix` function and *tf* $\times$ *IDF* terms weighting scheme; sparse keywords, i.e. appearing in less than 10% of documents are removed from the document-term matrix resulting in 125 keywords left in the representation of the documents; the vectors representing documents are normalized in such a way that each coordinate is divided by the norm of the vector; finally, the data matrix is turned into a data frame which is an R data structure more suitable for carrying out the classification task;
7) category number, case number and position in the case is assigned to each document/row in the above mentioned data frame;
8) the set of documents is divided into the training and testing parts as described earlier and that completes our data collection preparation.

B. Results

We have executed several runs of our algorithm. In each run the collection of documents has been divided randomly into the training and test data sets as described earlier.

In each experiment we executed 100 runs of the algorithm. In each run 20% of the cases was treated as on-going cases and only the documents at the cut-off points in these cases were classified. We repeated the same procedure for 30%, 40% up to 50% of cases treated as on-going.

Table I shows the obtained results in terms of the fraction of properly assigned categories (the second column) and the fraction of the properly assigned cases (the third column). These are mean values computed for 100 runs. The standard deviations are given, respectively, in column 4 and column 5. Each row corresponds to different percentage of the training/test cases. It should be noted that the assignment of the case is performed assuming that the category has been properly assigned first. Clearly, only a fraction of the results obtained is presented due to a lack of space.

TABLE I
AVERAGED ACCURACY OF CLASSIFICATION FOR 100 RUNS OF THE ALGORITHM ON RANDOMLY DIVIDED DATA SET (20% TO 50% CASES ARE THE ON-GOING CASES, ONLY CUT-OFF DOCUMENTS ARE CLASSIFIED

	Avg category accuracy	Avg case accuracy	Std dev for category	Std dev for case
20%	0.5986	0.6190	0.1177	0.2174
30%	0.5339	0.6439	0.0513	0.0773
40%	0.5478	0.5924	0.0474	0.0743
50%	0.5465	0.5655	0.0381	0.0623

We have executed another series of runs comparing the algorithm proposed here with two variants of the 1-nn algorithm: a "two-stage" variant and a "one-stage" variant. In the former, it is again assumed that the category has been properly recognized and the training data set is the set of all documents in all on-going cases available at a given moment. Thus, at the beginning the training data set is composed of all documents belonging to the cases of given category and preceding the cut-off points (cf. section IV-A). From this point of view, the algorithm proposed in this paper may be seen also as 1-nn algorithm but working with the training set comprising only the last documents (i.e., appearing just before the cut-off points) of cases belonging to a given category. As we intended to compare specifically the two-stage procedure with the one-stage one we also implemented the second variant of the 1-nn algorithm. It does make sense to take into account a direct one-stage version of our algorithm, i.e., such that a new document is compared to the last documents of all on-going cases from considered categories - such an algorithm have to give worse results or, at least, cannot produce better results. If it would indicate a case from the proper category then it have to be exactly the same case as indicated by two-stage version, and if it is a case from different category then it is surely wrong decision. Thus, we implemented a one-stage 1-nn algorithm which uses the training data comprising documents from all cases - up to the cut-ff point - and from all categories. This is a more fair comparison as this algorithm can classify a document to a proper case thanks to the access to all documents preceding the cut-off point in a proper case and not only to the last one as in case of our algorithm.

We show the results in Table II which is laid out in a similar way to the Table I. The first column, for completeness, shows the effectiveness of the classification to categories, next columns show the results for a two-stage algorithm proposed in this paper, 1-nn one-stage algorithm and 1-nn two-stage algorithm, respectively. Our simple algorithm gives better results than one-stage 1-nn but worse than two-stage 1-nn. However, it should be stressed that the analysis of particular classification decisions shows that our algorithm quite often makes better decisions the the latter one, i.e., it is not the case that the set of documents properly classified by our algorithm is just a subset of a set of documents properly classified by the latter algorithm.

The results of the experiments are promising. The novel two-stage classification algorithm adopted in the paper proves to be effective and efficient for the data set under consideration. The specificity of cases being composed of the subsequent sections of an article makes it possible to identify the proper case quite well by the comparison of the classified document with just the last document of the case, provided the category of document is properly identified first. This may not work for, e.g., a sequence of documents related to executing a tender as documents which occur as neighbors in such a sequence may be quite different. Thus, for the general task of the multiaspect text categorization it may be worthwhile to have at hand a number of algorithms and apply one which is appropriate for a given type of data. We are working on a more formal characterization of various types of data.

TABLE II
AVERAGED ACCURACY OF CLASSIFICATION FOR 100 RUNS OF THREE ALGORITHMS: TWO-STAGE PROPOSED IN THE PAPER, ONE-STAGE 1-NN AND TWO-STAGE 1-NN, ON A TEST DATA SET WITH 50% CASES RANDOMLY CHOSEN AS ON-GOING CASES; CUT-OFF DOCUMENTS AND ALL FOLLOWING THEM IN THE ON-GOING CASES ARE CLASSIFIED

	Category accuracy	Alg 1 accuracy	Alg 2 accuracy	Alg 3 accuracy
Mean	0.5483	0.5565	0.4772	0.6498
Std. dev.	0.0419	0.0274	0.0294	0.0284

V. CONCLUSION

We have defined and discussed the problem of multiaspect text categorization (MTC) in which textual documents have to be classified along different schemes, referred to as a set of categories and a set of cases. We proposed a novel two-stage algorithm which first assigns the document to a category and then to cases which belong to this category. Both stages are based on the well known and reliable k-NN algorithm. The algorithm has been preliminarily tested on a collections of documents which has been created on the basis of the ACL ARC collection. The first results are promising. It is expected that the algorithm may be effective and efficient for a specific type of data, in particular when subsequent documents in cases are closely related in terms of the used vocabulary. This is part of a broader work on the MTC problem in which we have developed some methods attempting to grasp the more elaborate relations between the documents in sequences as well as methods similar to the one proposed in this paper. The work is in progress and, in particular, test on larger corpora are under way.

ACKNOWLEDGMENT

This work is supported by the National Science Centre (contract no. UMO-2011/01/B/ST6/06908).

REFERENCES

[1] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Survys*, vol. 34, no. 1, p. 147, 2002.

[2] S. Zadrożny, J. Kacprzyk, M. Gajewski, and M. Wysocki, "A novel text classification problem and its solution," *Technical Transaction. Automatic Control*, vol. 4-AC, pp. 7–16, 2013.

[3] S. Zadrożny, J. Kacprzyk, and M. Gajewski, "A novel approach to sequence-of-documents focused text categorization using the concept of a degree of fuzzy set subsethood," in *Proceedings of the Annual Conference of the North American Fuzzy Information processing Society NAFIPS'2015 and 5th World Conference on Soft Computing 2015, Redmond, WA, USA, August 17-19, 2015*, 2015.

[4] A. S. Weigend, E. D. Wiener, and J. O. Pedersen, "Exploiting hierarchy in text categorization," *Inf. Retr.*, vol. 1, no. 3, pp. 193–216, 1999.

[5] M. Ceci and D. Malerba, "Classifying web documents in a hierarchy of categories: a comprehensive study," *J. Intell. Inf. Syst.*, vol. 28, no. 1, pp. 37–78, 2007.

[6] D. Koller and M. Sahami, "Hierarchically classifying documents using very few words," in *Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997), Nashville, Tennessee, USA, July 8-12, 1997*, D. H. Fisher, Ed. Morgan Kaufmann, 1997, pp. 170–178.

[7] S. Zadrożny, J. Kacprzyk, M. Gajewski, and M. Wysocki, "A novel text classification problem and two approaches to its solution," in *Proceedings of the International Congress on Control and Information Processing 2013 (ICCIP'13)*. Cracow University of Technology, 2013.

[8] M. J. Zaki, "SPADE: an efficient algorithm for mining frequent sequences," *Machine Learning*, vol. 42, no. 1/2, pp. 31–60, 2001.

[9] J. Kacprzyk and S. Zadrożny, "Power of linguistic data summaries and their protoforms," in *Computational Intelligence Systems in Industrial Engineering*, ser. Atlantis Computational Intelligence Systems, C. Kahraman, Ed. Atlantis Press, 2012, vol. 6, pp. 71–90.

[10] D. Olszewski, J. Kacprzyk, and S. Zadrożny, "Asymmetric k-means clustering of the asymmetric self-organizing map," in *Artificial Intelligence and Soft Computing*, ser. Lecture Notes in Computer Science, L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. Zadeh, and J. Zurada, Eds. Springer International Publishing, 2014, vol. 8468, pp. 772–783.

[11] J. Kacprzyk, J. W. Owsiński, and D. A. Viattchenin, "A new heuristic possibilistic clustering algorithm for feature selection," *Journal of Automation, Mobile Robotics & Intelligent Systems*, vol. 8, no. 2, 2014.

[12] D. Olszewski, J. Kacprzyk, and S. Zadrożny, "Time series visualization using asymmetric self-organizing map," in *Adaptive and Natural Computing Algorithms*, ser. Lecture Notes in Computer Science, M. Tomassini, A. Antonioni, F. Daolio, and P. Buesser, Eds. Springer Berlin Heidelberg, 2013, vol. 7824, pp. 40–49.

[13] M. Rybnik and W. Homenda, "A harmonization model with partial fuzzy knowledge," in *IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), 2013 Joint*, June 2013, pp. 1366–1371.

[14] M. Szymczak, S. Zadrożny, A. Bronselaer, and G. D. Tré, "Coreference detection in an XML schema," *Information Sciences*, vol. 296, pp. 237 – 262, 2015.

[15] J. Allan, Ed., *Topic Detection and Tracking: Event-based Information*. Kluwer Academic Publishers, 2002.

[16] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang, "Topic detection and tracking pilot study: Final report," in *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[17] M. Gajewski, J. Kacprzyk, and S. Zadrożny, "Topic detection and tracking: a focused survey and a new variant," *Informatyka Stosowana*, to appear.

[18] R. A. Baeza-Yates and B. A. Ribeiro-Neto, *Modern Information Retrieval - the concepts and technology behind search, Second edition*. Pearson Education Ltd., Harlow, England, 2011.

[19] L. Rabiner, "A tutorial on HMM and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[20] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, pp. 513–523, 1988.

[21] S. Bird, R. Dale, B. Dorr, B. Gibson, M. Joseph, M.-Y. Kan, D. Lee, B. Powley, D. Radev, and Y. Tan, "The ACL anthology reference corpus: A reference dataset for bibliographic research in computational linguistics," in *Proc. of Language Resources and Evaluation Conference (LREC 08)*, Marrakesh, Morocco, pp. 1755–1759.

[22] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2014. [Online]. Available: http://www.R-project.org

[23] I. Feinerer, K. Hornik, and D. Meyer, "Text mining infrastructure in R," *Journal of Statistical Software*, vol. 25, no. 5, pp. 1–54.