The Problem of First Story Detection in Multiaspect Text Categorization

Sławomir Zadrożny, Janusz Kacprzyk and Marek Gajewski

Abstract The new concept of *multiaspect text categorization* (MTC), recently introduced in a series of our papers, may be viewed as a combination of the classic and well-known text categorization (TC) and some kind of sequential data classification. The first aspect of the problem, i.e., the assignment of a document to a *category*, may be addressed using one of the well-known techniques such as, e.g., the *k*-nearest neighbors method. The second aspect is, however, less standard and boils down to the assignment of a document to one of the sequences, called *cases*, of documents maintained within a category. Cases cannot be treated in the same way as categories as, first, they contain an ordered—by the time of arrival—set of documents, and second, they are usually represented in a training dataset by a (relatively) small number of documents. Moreover, it is assumed that new cases can emerge during the document collection lifetime. Hence, the assignment of a document to a case is a challenging task by itself, and then the deciding if a document starts a new case is even more difficult. In this paper, we deal with the latter problem, discussing it in the broader perspective of sequential data mining and comparing a number of approaches to solve it.

1 Introduction

Text categorization (TC) [1] is among the most important tasks defined in the framework of the class (textual) information retrieval (IR) [2]. It plays an important role in the automatic handling of large document collections as it, basically, boils down

S. Zadrożny (✉) · J. Kacprzyk · M. Gajewski
Systems Research Institute, Polish Academy of Sciences,
ul. Newelska 6, 01-447 Warszawa, Poland
e-mail: Slawomir.Zadrozny@ibspan.waw.pl

J. Kacprzyk
e-mail: Janusz.Kacprzyk@ibspan.waw.pl

M. Gajewski
e-mail: Marek.Gajewski@ibspan.waw.pl

P. Kulczycki et al. (eds.), *Information Technology and Computational Physics*,
Advances in Intelligent Systems and Computing 462,
DOI 10.1007/978-3-319-44260-0_1

to the assignment of documents to some predefined categories. There are many different variants of this basic task which may be distinguished depending, e.g., on following aspects (cf., e.g., [1]). Documents to be classified may be available all at once (off-line categorization) or they are to be classified one by one when they appear on the input of the system (on-line categorization). The structure of categories may be flat (one level) or there may be a hierarchy of them (hierarchical text categorization). Each document may be assigned to at most one category (single-label categorization) or to many categories (multi-label categorization). Analogously to the general task of classification, a special case is when there are just two categories among which usually one is of interest (single-class categorization)—this may be confronted with the general case when the number of categories is a larger number (multi-class categorization), of course of a reasonable value, for comprehensiveness. The very nature of categories also makes a difference: a canonical variant of text categorization refers to the thematic (topical) categories while in other cases there may be essentially different criteria deciding on the grouping of documents into categories (e.g., the type of a business document in a company meant as one of the memo, advertisement brochure, meeting announcement etc.) Finally, the text categorization may be carried out: manually by an expert or a group of experts (an option viable only for small document collections); automatically, using some hand crafted rules (in the vein of knowledge engineering) or the whole process can be fully automated, i.e., some machine learning techniques can be used to automatically derive classifiers based on the set of training data.

In a number of papers [3–9] we have introduced the new concept of the *multiaspect text categorization* (MTC) and some approaches to solve it. The MTC problem may be seen as a special case of the general text categorization problem. Referring to the aspects of the TC mentioned earlier, it may be characterized as a TC problem which is on-line, hierarchical, single-label, multi-class, and with mixed types of categories, and for which a fully automatic solution is sought. The formulation of the MTC is motivated by a practical problem of managing collections of documents dealt with within an organization, notably a public institution in Poland, which has to follow some formal legal regulations. Namely, on the first level, the documents have to be arranged according to a hierarchy of prespecified thematic/topical categories. On the second level, each document has to be assigned within its category to a sequence of documents, referred to as a *case*. The cases will usually correspond to some business processes carried out by a company. For example, the process of the purchase of some accessories will be usually initialized with a formal request from a department in need of them, which may be followed by a call for tender, in turn followed by the offers from prospective suppliers, etc. Thus, besides a hierarchy of thematic categories we have to deal with a different kind of hierarchy relating a thematic category and cases belonging to this category.

The classification of documents on the first level alone may be directly dealt with using techniques known in the classic *text categorization* (TC) [1]. The second-level classification is, however, more difficult. The cases may be considered as categories, similarly to the situation at the first level, but the problem is implied, basically, by a limited number of training documents representing such a category and, moreover,

by the fact that only a part of such categories (cases) is known in advance. Any incoming document may turn out to be initiating a new case. Thus, an important part of a successful solution to the MTC problem is the detection if this takes place. In this paper we deal exactly with this problem.

In the next section, we remind the formal statement of the multiaspect text categorization problem. Then, we review the related work. Next, we propose the use of a number of techniques to solve the problem of the first story detection and, finally, we report the results of the computational experiments aimed at comparing these techniques.

2 The MTC Problem

The multiaspect text categorization (MTC) problem may be illustrated with an example of a public administration institution dealing with various affairs. One of the aspects of its activity is a proper organization of documents concerning particular efforts and yielded in the course of a business process carried out by this institution. An example of such a business process may be arranging a public tender for the purchase of office equipment. The related documents include the announcement of the tender, offers incoming from companies responding to the tender and offering the equipment, minutes of meetings of a committee responsible for carrying out the process, etc. Usually, there is specified a list of *categories* of affairs which are dealt with. Sometimes these categories are arranged in a hierarchy (in this paper we assume a flat, one level, list of categories but an interested reader may consult our another paper [10] as well as the literature therein on the hierarchical text categorization). The accomplishment of an instance of a business process will be referred to as a *case* and the institution has to store together and in a proper order all documents related to a given case.

Thus, we consider the MTC problem in the following context. There are many *on-going* cases belonging to various categories and our aim is to build an automatic system which will assist a human operator in assigning a new incoming document to a proper case, i.e., to a case which is related (possibly to a high extent) to a business process instance to which this document actually belongs. We assume that the system takes into account only the content of the document and does not use, e.g., metadata accompanying this document. Such an assumption may be more or less justified in various practical scenarios but it guarantees a broader applicability of the designed system.

What is very important is the fact that, if it is justified, a new document can initiate a new instance of a business process and thus can originate a case of which it becomes the first document. In this paper, we are interested in finding a way to automatically decide if a new incoming document really starts a new case in view of its contents and the contents of all documents stored so far, and their organization in categories and cases.

Let us now formally describe the above characterized problem. We assume that a collection D of documents is given:

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

These documents are assigned to some predefined *categories* from the set C:

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

in such a way that each document $d \in D$ is assigned to exactly one category $c \in C$. The documents are further arranged within each category into sequences $\sigma \in \Sigma$, rank ordered with respect to the time of arrival, which are referred to as *cases*:

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

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

Again, each document $d \in D$ belongs to exactly one case $\sigma \in \Sigma$.

The goal is to build a system, using D as the training collection, which will support a human user in deciding how to add a new incoming document d^* to the collection D. Thus, a document d^* has to be assigned to a category $c \in C$ and to a case $\sigma \in \Sigma$ within this category.

Various strategies may be adopted to obtain a proper classification. A two-level approach may be applied in which, first, a category is assigned and then the case. The motivation is that the classification to a category may be relatively easier and the classic text categorization techniques should be effective and efficient enough to do this. Then, when a category is already selected, one can expect that it should be easier to assign the document to a proper case within this category. The reason is that local characteristic features of cases in a given category may be employed and, moreover, the number of candidate cases will be much lower in such a scenario; cf., e.g., our papers [6, 8, 9] for examples of such an approach in the framework of the MTC or the paper by Yang et al. [11] for a related approach in another context. It is worth noting that it is also possible to skip the assignment of a category to a document d^* and to focus on the choice of a proper case as such a choice directly implies also a category c to which the case σ belongs. However, this way the extra information on the category of the document d^* is ignored when choosing the case, provided that the category assignment is successful. Finally, the decisions concerning the assignment of d^* to a category and to a case may be combined with the hope that both decisions will mutually support each other. An example of such an approach is given in our paper [5].

To summarize, the MTC problem may be characterized as a text categorization problem with two levels of broadly defined categories. At the upper level, these may be assumed to be typical prespecified thematic categories, represented in the training collection d with a sufficiently large number of examples. At the lower level, these are cases the number of which is dynamically changing and which may be poorly, or even not at all, represented in the training collection of the documents, D.

3 Related Works

The task considered in this paper refers mainly to the context of the multiaspect text categorization problem (MTC) recently proposed in our earlier work; cf., e.g., [4], and formally presented in the previous section. A similar problem known in literature is the *Topic Detection and Tracking* (TDT) [12].

The 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 TDT started in 1997 [13] and was followed by regular workshops during the next 7 years. The topic of detection and tracking is considered in the context of processing of a stream of news coming from various sources and concerning some events/topics. It is assumed that events evolve over time and some new news stories related to them are incoming. However, new events are happening which are also represented in the stream of incoming news stories. The basic task is here to group together news stories concerning the same events and describing their development over time, various aspects etc.

An individual piece of news in TDT is referred to as a *story* and corresponds to a document in our new MTC problem definition. 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. A number of specific tasks are distinguished within the 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 set of groups of stories forming particular topics—which may be empty in the beginning—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. It is however distinguished due to its importance and difficulty [14].

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

1. categories and cases are considered in the MTC as opposed to topics only in the 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 the 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 earlier paper [7].

In the current paper, we are concerned with a counterpart of the first story detection (FSD) task present in our multiaspect text categorization problem. Thus, it is worthwhile to briefly review a few techniques that have been proposed to solve the FSD in the framework of the TDT.

Approaches to first story detection are often based on the similarity of the incoming document d^* with respect to all or a part of the documents collected earlier. A threshold is assumed and if the similarity of a document d^* with respect to, e.g.:

- any of the earlier collected documents, or
- any of k recently collected documents, or
- any centroid of documents belonging to particular topics earlier recognized,

exceeds the assumed threshold, then document d^* is deemed to be related to some earlier seen topic and is assigned to it. Otherwise it is treated as starting a new topic and becomes its first story.

Another idea consists in the monitoring of term distribution over time and a new topic is recognized in case an abrupt change in this distribution is detected for a given story. Another strategy in this vein refers to the solution of another TDT task, namely that of *topic tracking*. This task consists in deciding if an incoming story d^* belongs to a given topic represented by a (small) number of stories. Assuming that the tool for topics tracking is available it may be immediately used to solve also first story detection problem. Namely, if an incoming document d^* is not indicated as belonging to any tracked topic then it has to be the first story of a new topic.

Yang et al. [11] propose a relatively simple, and effective and efficient approach to the FSD problem which is, moreover, very relevant to our MCT problem and the detection of the first document of a new case. Namely, they group the topics into a higher level of categories (originally, in [11], topics are referred to as events and categories as topics but we will keep here the terminology consistent with the previously introduced one for the TDT problem). An incoming story is first classified to a category and only then to a topic within that category. If the latter classification fails, i.e., the similarity of a new document to its closest neighbor within given category is lower than some threshold value, then such a story is qualified as a first story of a new topic. This resembles very much our two-level approach to assigning a document to a case within an earlier chosen category; cf., e.g., our [6]. The motivation for the Yang et al. [11] approach is that it makes it possible to use different features (keywords) while classifying a story to a category and to a topic. Thus, the features shared by first stories of topics with all other stories belonging to a given category may be taken into account while classifying a document to a category but they may be ignored (treated as stopwords) while classifying this document to a topic. Thanks to that, an actual first story may be properly recognized as such because it may turn out not to be similar to other stories in a given category even if it shares a number of features with them. The important points of this approach are the following:

- different representation of stories for their classification at the levels of categories and topics via a separate feature selection at each level;
- enhancing the basic vector space model representation of stories with named entities.

For the top-level classification of the stories Yang et al. [11] use the Rocchio-style classifier, popular in the framework of the TDT [15]. The recognition of the first story at the lower level of classification is based on the similarity to a nearest neighbor, as mentioned earlier.

The problem of first story detection may be considered in a broader framework of the *novelty detection* problem [16]. The concept of novelty detection, in general, refers to recognizing that an object under consideration belongs to a class which has not yet been represented in a training dataset. In the MTC context we face this problem in an even more intense form as if we treat cases as classes then:

- evidently, we should expect incoming documents belonging to new cases, i.e., new classes, and moreover,
- even for cases (classes) represented in the training dataset, very often this representation will be very limited, i.e., a case will be often 1–2 document long.

The novelty detection approaches address directly the first of the above problems but usually take into account also the sparsity of the training data in which the examples of novel data are scarce.

The statistical approaches to novelty detection often consider the problem as a binary classification task aiming at distinguishing novel data from the rest, "normal" data [16, 17]. Popular solutions are based on estimating the probability density for the data belonging to the "normal" class and deciding on the novelty of incoming data objects if they fall in regions of low density. Examples of the approaches in this vein, specifically meant for the novelty detection in the textual information processing context, include those presented in the papers by Hofmann et al. or Hansen et al. [18, 19]. Recent surveys on novelty detection are papers by de Faria et al. [17, 20].

Some other related concepts discussed in the literature are also relevant for the FSD problem definition and solution. These include anomaly detection and rare events mining; cf., e.g., [21]. This is due to the fact that if a standard binary classification approach is adopted to solve the FSD problem, then one class, of the first stories, will be usually an order of magnitude smaller than another class of non-first stories.

Our task may also be studied from a broader perspective of applying machine learning methods to the sequential data [22]. Namely, the main idea of an intelligent approach to classifying a document to a proper sequence calls for understanding the mechanism behind the forming of document sequences within a given collection or a part of it (category). Knowing this mechanism, we can decide if a document under consideration fits an existing sequence or rather should be treated as starting a new sequence. In our earlier works [4, 23], we propose to employ, first of all, tools and technique of the hidden Markov models (HMMs) to get such an understanding of sequences of documents within categories. Hidden states may then be identified with stages of a business process which produce a given sequence of documents (a case). If these stages may be explicitly identified, then a broader repertoire of models/techniques for sequential data processing may be considered as helpful [22] such as, e.g., the conditional random fields (CRF).

4 A Direct Approach for Solving the FSD as a Classification Problem

In our approach reported in this paper we adopt an approach to the FSD task solving differently from most of those proposed in the framework of the TDT. Namely, the latter approaches employ a topic tracking technique and declare a story as a first story when it does not fit any of the topics recognized so far; cf. Sect. 3. This observation applies also to the approach proposed by Yang et al. [11], even if it introduces a two-level classification schema. Our approach is an attempt to solve the FSD problem using directly a binary classification. Thus, we start with a collection of training documents which are organized in cases (sequences) according to the definition of the MTC problem. The first documents of all cases present in the collection are positive examples while the remaining documents form the set of negative examples. Then, we employ some variants of a number of well-known machine learning algorithms and compare their effectiveness.

The algorithms we started with are the following:

1. a variant of the *k*-nearest neighbors algorithm,
2. the random forests,
3. logistic regression,
4. linear discriminant analysis,
5. an approach based on modeling the probability density of selected keywords in positive and negative examples,
6. support vector machine.

We have also tested a feature selection technique as well as two schemes of training the classifiers:

- locally, a separate individual classifier for each category,
- globally, one classifier for the whole collection.

In the tests we carried out, the feature selection techniques did not improve the results for most of the considered algorithms. On the other hand, most of the algorithms produced much better results for the global variant mentioned above, i.e., when one classifier is constructed for recognizing first stories based on the whole training collection. Thus, in the next section we report the results of our experiments only for the case where stories are represented using all considered features (keywords) and for the global approach. Now, we will briefly describe the algorithms and justify their use in our experiments.

The *k*-nearest neighbors technique (*k*-nn)-based algorithms proved to be effective and efficient in our earlier approaches to the MTC problem. In [6, 9] we use a variant of *k*-nn to assign a category and a case to a document. It is also widely used by the TDT community (cf., e.g., [24]). We use it here in the version proposed by Yang et al. [24] (cf. also [9]) which is referred to as kNN.avg2. It is based on a function defined as follows:

$$r(d^*, k_p, k_n, D) = \frac{1}{|U_{k_p}|} \sum_{d \in U_{k_p}} sim(d^*, d) - \frac{1}{|V_{k_n}|} \sum_{d \in V_{k_n}} sim(d^*, d) \quad (5)$$

The value of this function is computed for the document d^* to be checked for being a first story with respect to the training dataset D. There are two parameters k_p and k_n which determine the cardinality of the sets U_{k_p} and V_{k_n}, respectively. The former set comprises the k_p positive examples (i.e., first stories in the training data set D) most similar to the document d^* while the latter set comprises k_n negative examples (i.e., non-first stories in the training data set D) most similar to d^*. In our experiments we use the kNN.avg2 algorithm with the parameters set as follows: $k_p = k_n = 1$. The similarity is computed using a function denoted by *sim* which is identified with the classic cosine measure in [24] and with the complement to the Euclidean distance in [9] (vectors representing documents are assumed to be normalized and, thus, there is a well-defined maximal possible Euclidean distance between two documents). Thus, the value of the function r for a document d^* is its average similarity to k_p most similar first stories reduced by its average similarity to the k_n most similar non-first stories. If there are less than k_p positive documents in D then all positive documents in D are employed. The same applies to the negative documents (even if this can rarely happen).

The document d^* is recognized as the first story if:

$$r(d^*, k_p, k_n, D) > 0$$

and as the non-first story otherwise, for the chosen values of the parameters k_p and k_n. As compared to the standard k-nn technique, the kNN.avg2 version is more suitable for the first story detection problem solving as it addresses the problem of imbalance between the positive and negative classes (usually there will be much less first stories than non-first stories in the training data set) using a fixed number of the nearest positive and negative examples. At the same time it also takes into account how similar the nearest examples actually are.

The algorithm of random forests [25] is used in the experiments in the version implemented as the `randomForest` function in the package `randomForest` using standard parameters. In particular, the number of trees to grow is set to 500. The model is constructed to distinguish first stories using all keywords used to represent the documents in the collection. This is one of the algorithms deemed to be highly effective and efficient in the machine learning community. It has a sophisticated built in mechanism for feature selection which should help recognize the first stories which, as argued earlier, are by definition very similar to other documents in a given category but are expected to be less similar with respect to some subset of specific keywords.

The logistic regression algorithm [26] is used in the experiments using the `glm` standard function of the R environment. The binomial distribution over the positive (first stories) and negative (non-first stories) classes is modeled via the logit link function using again the weights of all keywords as independent variables in the

linear regression analysis. This algorithm also belongs to the most popular discriminative classification techniques [27].

The fourth algorithm employed is the one based on linear discriminant analysis [28]. One of the classic techniques which, basically, requires class conditional normal distributions. In our case, the multidimensional distributions of the documents characterized by keywords weights are far from normal within the classes of both the first stories and non-first stories due to, e.g., a high sparsity of the document-term matrices. However, this technique is known to be robust and in our computational experiments it has also proved to be good.

The fifth algorithm used is a simple attempt to apply an aggressive dimension reduction technique combined with a straightforward probabilistic approach which boils down to the naive Bayes approach with the kernel density estimation [28]. Namely, first a number of keywords $t \in T$ with the highest mean in the representations of the first stories present in a training dataset are selected. Then, those whose mean is significantly higher than in the non-first stories are preserved and form a set $T_s \subset T$. The t-test is used to assess the significance with p-value equal 0.05. Their standard deviation in the first stories is also recorded. Then, two probability density functions are constructed using a kernel-based method [29], for each of these keywords: in the first stories and in the non-first stories of the training dataset. Then, the following function is used to discriminate first stories from non-first stories:

$$g(d) = \log(\frac{P(fs|d)}{P(nfs|d)}) = \\ = \log(\frac{P(fs)}{1 - P(nfs)}) + \sum_{t \in T_s} (\log(f^t_{fs}(d[t])) - \log(f^t_{nfs}(d[t])) \quad (6)$$

where $d[t]$ denotes the weight of a keyword t in the representation of a document d, f^t_{fs} and f^t_{nfs} are approximated conditional probability density functions of the particular keywords $t \in T_s$ in the first stories and non-first stories of the training dataset, respectively, and $P(fs)$, $P(nfs)$ are a priori probabilities of a document being a first story and non-first story, respectively. The latter a priori probabilities are estimated on the training data set. However, in our experiments reported in Sect. 5 the assumption of the a priori probability equal 0.5 for both classes produced much better results and, thus, we adopted this strategy. The function $g(d)$ corresponds therefore to the logarithm of the odds of a document to be the first story, assuming the conditional independence of the keywords in documents of both classes. Thus, if $g(d) > 0$, then the document d is classified as the first story and, otherwise, as the non-first story.

A variant of formula (6) is also employed:

$$g'(d) = \sum_{t \in T_s} \sigma'(t)(\log(f^t_{fs}(d[t])) - \log(f^t_{nfs}(d[t])) \quad (7)$$

where $\sigma'(t) = 1 - \frac{\sigma(t)}{\max_s \sigma(s)}$, $\sigma(t)$ denotes the standard deviation of the weights of keywords in the first stories of the training dataset. This variant is meant to differentiate

the influence of particular keywords $t \in T_s$ on the classification decision, i.e., the keywords with a higher standard deviation have a lower influence. In the formula (7) we assume the a priori probabilities equal 0.5, as mentioned earlier, and thus here we dropped the first component of the formula (6).

The purpose of this approach is a direct selection of keywords that are relatively highly frequent in the first stories and less frequent in the non-first stories. In our experiments, it was most often possible to spot such keywords. In case it was not possible, all keywords were taken into account. The approach is similar to the linear discriminant analysis (LDA) but explicitly drops the assumption on the normality of distributions required by the LDA.

Finally, the sixth algorithm employed is a powerful and very popular support vector machine (SVM) method [30]. We experimented with various kernels and other parameters and finally decided to use the RBF kernel. The SVM technique perfectly fits in its original form the binary problem of distinguishing first stories from non-first stories.

5 Computational Experiments

We have tested the approaches presented in the previous section using a data collection which we have used also in our previous work [5, 6, 8, 9]. The starting point is the set of articles on computational linguistics available in the framework of the ACL Anthology Reference Corpus (ACL ARC) [31]. We use a subset of 113 papers. The papers are originally partitioned into sections and the idea is to treat each article as a case and its sections as documents of such a case. What is missing is the grouping of documents/cases into categories. Thus, to do this, we first use the k-means algorithm to partition the set of articles into 7 clusters which play the role of categories. This number of clusters has been chosen experimentally in order to secure a reasonable number of categories and their cardinalities.

The articles and, later on, the resulting documents are represented using the vector space model (cf., e.g., [2]) and standard preprocessing techniques such as the removal of the punctuation, numbers, and multiple white spaces; stemming; changing all characters to the lower case; dropping stopwords and words shorter than 3 characters. The $tf \times IDF$ scheme is employed to compute the weights of particular keywords in documents. The obtained document-term matrix is very sparse and, thus, the keywords present in less than 10% of the papers are further removed. As a result 125 keywords are employed. The vectors representing particular documents are normalized by dividing each coordinate by the Euclidean norm of the whole vector and thus the Euclidean norm of each vector equals 1.

Finally, we obtain a collection of 113 cases comprising 1453 documents which is then split into the training and testing datasets. A number of cases are randomly chosen and a cut-off point in each of them is again randomly selected. All documents at positions starting from the cut-off point are removed from the case and form the test dataset (in the experiments reported here we used the datasets composed of only

the documents located at the cut-off points). All remaining documents from the collection serve as the training dataset. This way, if a cut-off point corresponds to the first position in a case we obtain a first story in the test data set.

All computations are carried out using the R platform [32] with the help of the packages: `tm` [33], `FNN` [34], `randomForest` [35], `kernlab` [30], `MASS` [36] and our own R scripts.

We have tested the algorithms for first story detection mentioned in Sect. 4 in two configurations:

1. for each category separately, i.e., we assumed that the incoming document d^* has been first properly assigned a category and only then it is checked as a candidate for being a first story based on the training data set confined to this category; this is the case referred by Yang et al. as the *simple case* (cf Table 2 in [11];
2. for the whole collection at once, i.e., the incoming document d^* is first checked for being a first story using the whole training data set; this is referred as the *baseline* case in [11].

All algorithms have proved to give better results for the second configuration. We have expected that for different categories different keywords may be better at distinguishing between first stories and non-first stories. However, this potential cannot be exploited due to a limited number of training documents representing first stories when considered for each category separately. The similar conclusions follow from the experiments reported in [11], even if a slightly different context of the TDT is considered there. Thus, in what follows we will present the results only for the second strategy.

We have run a series of 200 experiments and their results are presented in Table 1. In each run we randomly select 56 cases (i.e., 50% of all cases) as on-going, i.e., those in which randomly a cut-off point is selected as earlier described. In order to evaluate the results obtained using particular approaches we use the F1 measure and the cost-based measure CO_{fsd} (in its normalized version) employed by Yang et al. [11]. These

Table 1 The results of 200 runs of the compared algorithms given in terms of the F1 and CO_{fsd} measures. The mean values and standard deviations of both measures are reported. Notice that for the second measure lower values indicate better results

The algorithm	F1		CO_{fsd}	
	Mean	sd	Mean	sd
k-nn	0.1705	0.1349	0.9912	0.2425
Random forests	0.2319	0.2278	0.9426	0.1890
Logistic regression	**0.4063**	0.1847	**0.6789**	0.2483
Linear discriminant analysis	**0.4427**	0.2094	**0.6571**	0.2684
Naive Bayes with kernel density estimation	0.26	0.1727	0.8735	0.2320
As above with standard deviation based keywords weighting	0.3384	0.1730	0.7356	0.2636
Support vector machine	0.3441	0.2195	0.8169	0.2407

measures are defined as follows, denoting the elements of the standard contingency table as TP, FP, TN, and FN, i.e., the true positives, false positives, true negatives, and false negatives, respectively:

$$F1 = \frac{2 * TP}{2 * TP + FP + FN} \tag{8}$$

$$CO_{fsd} = \frac{CO_m * P_m * P_t + CO_f * P_f * P_{nt}}{\min(CO_m * P_t, CO_f * P_{nt})} \tag{9}$$

where CO_m and CO_f are costs of the *miss* and *false alarm*, respectively, i.e., the former is the cost of classifying a first story as the non-first story while the latter is the cost of classifying a non-first story as the first story; $P_m = \frac{FN}{TP+FN}$ and $P_f = \frac{FP}{TN+FP}$, i.e., are the *false negative rate* (*miss*) and the *false positive rate* (*fall-out*), respectively; P_t and $P_{nt} = 1 - P_t$ are probabilities of a first story and non-first story occurrence, respectively.

Thus, (9) expresses the expected cost of the error to be made by the first story detection system. It is normalized by the cost of the better of two trivial algorithms which would classify all stories as first stories or as non-first stories, respectively. We set $CO_m = 1.0$ and $CO_f = 0.1$ after [11], adopting the justification given there that a miss may be easier recognized as a mistake by the human operator assisted by our system. On the other hand, we set $P_T = 0.1$ for the whole collection and for each category separately as this is the average frequency of first stories therein.

The results shown in Table 1 indicate the linear discriminative analysis and logistic regression as the best algorithms in detecting first stories. Due to the Wilcoxon two-sided test the former is significantly better than the latter in terms of both the F1 and CO_{fsd} measures. The second group form the Algorithms 6 (naive Bayes with kernel density estimation and standard deviation based keywords weighting) and 7 (support vector machines). Both are not significantly different concerning their effectiveness in terms of F1 measure but the latter is better in terms of CO_{fsd} measure. The latter effect is due to a very high number of false alarms (false positives) produced by Algorithm 6 compared to Algorithm 7, even if the former produced also slightly more true positives than the latter.

Summarizing, the effectiveness of the best of the tested methods is not fully satisfactory but taking into account the well-known difficulty of the first story detection problem it is not that bad. Yang et al. [11] report better results in terms of the CO_{fsd} measure but for a different dataset and using a richer representation of documents.

The fact that we obtained the best results using linear discriminant analysis is interesting in itself. The method is based on a rather strong assumptions which are not satisfied in our experiments and is fairly simple at the same time. It could be expected that Algorithm 5 should better fit the problem in question. However, it turns out that it performs rather poorly and only if combined with an extra weighting of the keywords produces relatively as good results as Algorithm 6. It should be however noted that both Algorithms 5 and 6 operate on a highly reduced set of keywords.

6 Conclusion

We have addressed the crucial problem of first story detection (FSD) in the framework of the multiaspect text categorization (MTC), a new problem class introduced in our former papers. The adopted approach is a rather straightforward one and boils down to formulating the FSD as a classic binary classification problem. Then, we have employed a number of standard classification algorithms, proposing some extensions in case of some of them. The best results have been obtained using the standard linear discriminant analysis. It should be noted that we have used a simple vector space model based representation of the documents. Our conclusions are based on computational experiments carried out on a dataset used in our previous work. Thus, our plans for a further research comprise both the search for a more sophisticated documents representation and more extensive tests on larger and more numerous datasets.

Our approach, though relatively straightforward and intuitively appealing, is still not that popular in the context of topic detection and tracking (TDT) in which the FSD problem is quite similar to the FSD considered in the context of our MTC problem. The approaches proposed by the TDT community usually base their approaches to the FSD problem on the same algorithm which is used for the TDT. Namely, a document is classified as the first story if it does not qualify as belonging to one of the recognized topics so far. The latter decision is in turn based on checking its similarity to the previously seen documents or their representatives (e.g., centroids of the documents related to the same topic) against some threshold value. Such an approach has been failing so far in case of our algorithms for the MTC problem and that is the motivation for our search for another solution.

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

References

1. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. **34**(1), 1–47 (2002)
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. ACM Press and Addison Wesley (1999)
3. Zadrożny, S., Kacprzyk, J., Gajewski, M., Wysocki, M.: A novel text classification problem and two approaches to its solution. In: Proceedings of the International Congress on Control and Information Processing 2013. Cracow University of Technology (2013)
4. Zadrożny, S., Kacprzyk, J., Gajewski, M., Wysocki, M.: A novel text classification problem and its solution. Tech. Trans. Autom. Control 4-AC, 7–16 (2013)
5. Zadrożny, S., Kacprzyk, J., Gajewski, M.: 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)

6. Zadrożny, S., Kacprzyk, J., Gajewski, M.: A new two-stage approach to the multiaspect text categorization. In: IEEE Symposium on Computational Intelligence for Human-like Intelligence, CIHLI 2015, Cape Town, South Africa, December 8–10, 2015. IEEE 2015, pp. 1484–1490 (2015)
7. Gajewski, M., Kacprzyk, J., Zadrożny, S.: Topic detection and tracking: a focused survey and a new variant. Informatyka Stosowana **2014**(1), 133–147 (2014)
8. Zadrożny, S., Kacprzyk, J., Gajewski, M.: A new approach to the multiaspect text categorization by using the support vector machines. In: De Tré, G., Grzegorzewski, P., Kacprzyk, J., Owsiński, J.W., Penczek, W., Zadrożny, S. (eds.) Challenging problems and solutions in intelligent systems, pp. 261–277. Springer International Publishing, Heidelberg (2016)
9. Zadrożny, S., Kacprzyk, J., Gajewski, M.: Multiaspect text categorization problem solving: a nearest neighbours classifier based approaches and beyond. J. Autom. Mob. Rob. Intell. Syst. **9**, 58–70 (2015)
10. Zadrożny, S., Kacprzyk, J., Gajewski, M.: A hierarchy-aware approach to the multiaspect text categorization problem. In: Proceedings of the World Conference on Soft Computing, Berkeley, CA, US (2016, in press)
11. Yang, Y., Zhang, J., Carbonell, J., Jin, C.: Topic-conditioned novelty detection. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: ACM, pp. 688–693 (2002)
12. Allan, J. (ed.) Topic Detection and Tracking: Event-based Information. Kluwer Academic Publishers (2002)
13. Allan, J., Carbonell, J., Doddington, G., Yamron, J., Yang, Y.: Topic detection and tracking pilot study: final report. In: Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop (1998)
14. Allan, J., Lavrenko, V., Jin, H.: First story detection in TDT is hard. In: Proceedings of the Ninth International Conference on Information and Knowledge Management, CIKM '00, pp. 374–381. ACM, New York, NY, USA (2000)
15. Yang, Y.: An evaluation of statistical approaches to text categorization. Inf. Retriev. **1**(1–2), 69–90 (1999)
16. Markou, M., Singh, S.: Novelty detection: a review—part 1: statistical approaches. Signal Process. **83**(12), 2481–2497 (2003)
17. De Faria, E., Gonçalves, I., Gama, J., De Leon Ferreira Carvalho, A.: Evaluation of multiclass novelty detection algorithms for data streams. IEEE Trans. Knowl. Data Eng. **27**(11), 2961–2973 (2015)
18. Hofmann, D.B.T., Baker, L.D., Hofmann, T., Mccallum, A.K., Yang, Y.: A hierarchical probabilistic model for novelty detection in text (1999)
19. Hansen, L.K., Sigurdsson, S., Kolenda, T., Nielsen, F.A., Kjems, U., Larsen, J.: Modeling text with generalizable gaussian mixtures. In: Proceedings of ICASSP'2000, pp. 3494–3497. IEEE (1999)
20. De Faria, E., Gonçalves, I., De Leon Ferreira Carvalho, A., Gama, J.: Novelty detection in data streams. Artif. Intell. Rev. **45**(2), 235–269 (2016)
21. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: a survey. ACM Comput. Surv. **41**(3) (2009)
22. Dietterich, T.G.: Machine learning for sequential data: a review. In: Caelli, T., Amin, A., Duin, R.P.W., Kamel, M.S., de Ridder, D. (eds.) Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshops SSPR 2002 and SPR 2002, Windsor, Ontario, Canada, August 6–9, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2396, pp. 15–30. Springer (2002)
23. Zadrożny, S., Kacprzyk, J., Gajewski, M.: A solution of the multiaspect text categorization problem by a hybrid HMM and LDA based technique. In: 16th International Conference Information Processing and Management of Uncertainty in Knowledge-Based Systems, Eindhoven, The Netherlands (2016, in press)
24. Yang, Y., Ault, T., Pierce, T., Lattimer, C.W.: Improving text categorization methods for event tracking. In: SIGIR, pp. 65–72 (2000)

25. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
26. McCullagh, P., Nelder, J.: Generalized Linear Models, 2nd edn. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis (1989)
27. Ng, A.Y., Jordan, M.I.: On discriminative vs. generative classifiers: a comparison of logistic regression and naive bayes. In: Dietterich, T.G., Becker, S., Ghahramani, Z. (eds.) Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3–8, 2001). Vancouver, British Columbia, Canada], pp. 841–848. MIT Press (2001)
28. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer Series in Statistics. Springer New York Inc., New York, NY, USA (2001)
29. Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Chapman and Hall, London (1986)
30. Karatzoglou, A., Smola, A., Hornik, K., Zeileis, A.: kernlab-an S4 package for kernel methods in R. J. Stat. Softw. **11**(9), 1–20 (2004)
31. Bird, S., et al.: The ACL anthology reference corpus: a reference dataset for bibliographic research in computational linguistics. In: Proceedings of Language Resources and Evaluation Conference (LREC 08), Marrakesh, Morocco, pp. 1755–1759
32. R Core Team: R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2014). http://www.R-project.org
33. Feinerer, I., Hornik, K., Meyer, D.: Text mining infrastructure in R. J. Stat. Softw. **25**(5), 1–54 (2008)
34. Beygelzimer, A., Kakadet, S., Langford, J., Arya, S., Mount, D., Li, S.: FNN: Fast Nearest Neighbor Search Algorithms and Applications, R package version 1.1 (2013). http://CRAN.R-project.org/package=FNN
35. Liaw, A., Wiener, M.: Classification and regression by randomforest. R News, vol. 2, no. 3, pp. 18–22 (2002). http://CRAN.R-project.org/doc/Rnews/
36. Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S, 4th edn. Springer, New York (2002)