A Solution of the Multiaspect Text Categorization Problem by a Hybrid HMM and LDA Based Technique

Sławomir Zadrożny(✉), Janusz Kacprzyk, and Marek Gajewski

Systems Research Institute, Polish Academy of Sciences,
ul. Newelska 6, 01-447 Warszawa, Poland
{zadrozny,kacprzyk,gajewskm}@ibspan.waw.pl

Abstract. In our previous work we introduced a novel concept of the *multiaspect text categorization* (MTC) task meant as a special, extended form of the text categorization (TC) problem which is widely studied in *information retrieval*. The essence of the MTC problem is the classification of documents on two levels: first, on a more or less standard level of thematic categories and then on the level of document sequences which is much less studied in the literature. The latter stage of classification, which is by far more challenging, is the main focus of this paper. A promising way of attacking it requires some kind of modeling of connections between documents forming sequences. To solve this problem we propose a novel approach that combines a well-known techniques to model sequences, i.e., the Hidden Markov Models (HMM) and the Latent Dirichlet Allocation (LDA) technique for the advanced document representation, hence obtaining a hybrid approach. We present details of our proposed approach as well as results of some computational experiments.

Keywords: Multiaspect text categorization · Sequences of documents · HMM · LDA

1 Introduction

We deal with a variant of the *text categorization* (TC) problem. In its basic form, the general TC problem boils down to deciding which of a predefined set of categories a given document belongs to. Thus, usually after adopting an appropriate document representation, e.g., based on the vector space model [2], documents are treated as vectors and one of a multitude of the classification techniques may be employed. In a series of papers [8,16,18,19] (see www.ibspan.waw.pl/~zadrozny/MTC for a complete list of our related papers). we have introduced and studied the concept of the *multiaspect text categorization* (MTC), a novel problem that goes far beyond the usual TC. We have proposed several approaches to solve it. The MTC task may be seen as a special case of the general text categorization problem where two levels of classification are involved. It is inspired by a practical application which may be briefly described as follows.

J.P. Carvalho et al. (Eds.): IPMU 2016, Part I, CCIS 610, pp. 214–225, 2016.
DOI: 10.1007/978-3-319-40596-4_19

Institutions in Poland, as well as in virtually all countries, are obliged to handle documents related to their business processes in a strictly regulated way. First, the documents have to be assigned to some thematic/topical categories arranged in a hierarchy. For example, a document submitted by a citizen while applying for a driving license should be classified as belonging to the top category "Social and civic cases" or, perhaps, within its specialized descendant subcategory at the bottom of a hierarchy, such as "Documentation of a vehicle registration". Second, within such a category this document has to be classified to a specific *case*, i.e., a sequence of documents related to a particular instance of the business process of the driving license issuing of that person. Such a sequence may already exist – for instance, the document under consideration may concern some additional information the applicant has been required to provide – or it may be the first document which initiates a case, e.g., it is the application for the issuing of the driving license of that person. Thus, within the case the documents are sequentially ordered and their order is implied by the logical succession of the documents within a given business process. Each instance of a given process may clearly be associated with a different number of documents of a different type, e.g., some documents may be initially missing and the institution will send a notice to the applicant to complete it which he or she will respond attaching those missing documents or explaining the reasons for their lack, or asking for further information from the institution etc.

The above task is usually dealt with manually, which is costly and time consuming, and our aim is to support the human operator by developing a system automatically generating an advice concerning the proper classification of documents. Thus, on the first level one may apply one of the classification techniques well studied in the classic *text categorization* [13]. The second level classification is more challenging due to several reasons. First of all, there is a limited number of training documents representing particular cases and a straightforward approach of treating each case as a category on its own does not work well. Moreover, the list of cases is growing over time and a classifier has to detect if a document to be classified should start a new case. Hence, grasping the logic of succession of the documents within a case seems to be critical for a classifier to successfully handle the MTC problem. In our previous work we proposed several solutions to the MTC problem. In particular, in [16] we proposed two approaches to model the sequences of the documents using Zaki's sequence mining algorithm [20] and the Hidden Markov Models [12]. In this paper we further develop the latter approach and go beyond the conceptual presentation by making the approach operational. In particular, we follow a widely advocated approach of developing a hybrid system that combines a variant of the well-known HMM technique with one of the modern techniques used to represent textual documents and known as the Latent Dirichlet Allocation (LDA) [4].

We first remind the formal definition of the MTC problem and point out some related works. Next, we present a general scheme of our proposed approach, briefly reminding the basics of the HMM and LDA techniques, focusing on their

hybridization. Then, we discuss details of our approach, present the results of some computational experiments and conclude with some final remarks.

2 The Multiaspect Text Categorization Problem

We assume a collection of documents, $D = \{d_1, \ldots, d_n\}$ which is structured as follows. The documents are arranged in a set of predefined *categories* from the set $C = \{c_1, \ldots, c_m\}$ in such a way the each document $d \in D$ belongs to exactly one category $c \in C$. The documents are further arranged within each category in sequences $\sigma \in \Sigma$ which are referred to as *cases*:

$$\sigma_k =< d_{k_1}, \ldots, d_{k_T} > \tag{1}$$

$$\Sigma = \{\sigma_1, \ldots, \sigma_p\} \tag{2}$$

Each document $d \in D$ belongs to exactly one case $\sigma \in \Sigma$. A different rationale and logic assumed for the grouping of the documents into categories and cases is here important. That is, respectively, a topical similarity or the belongingness to the same business process, in relation to its different stages.

Our purpose is to develop a system, following the paradigm of supervised learning, working in such a way that given a collection D structured as above and a new document $d^\star$ the system supports a human user in deciding how to assign $d^\star$ to a category $c \in C$ and to a case $\sigma \in \Sigma$ within this category. For practical reasons, we distinguish between *on-going cases* comprising documents of the business processes still under way, and closed cases related to the business processes which are already completed. The newly incoming documents may be classified only to the cases of the former type while cases of both types may serve as training examples for the construction of a classifier.

The MTC problem formulation and its practical inspirations are original and the literature of this topic basically comprises our recent works only. The most similar problem already known in the literature is *Topic Detection and Tracking* (TDT) [1]. It is inspired by a practical problem of handling a stream of news stories to be organized in a dynamically structured collection. News/documents concerning the same topic/event have to be grouped together and, similarly to our MTC problem, incoming documents may belong to already existing groups or may start new ones. Topics in the TDT are similar to the cases in our MTC problem and, in general, both problems share many points. However, they are inherently different which well justifies the study of the MTC as a separate problem. For example, in the TDT there are no such distinct two levels of classification as in the MTC. Even if the concept of a hierarchical TDT was also considered as an extension to the basic TDT, still the different nature of classes at particular levels of the hierarchy is not considered there. Another important aspect distinguishing both problems is that in the MTC cases are sequences of documents while topics in TDT are just sets of stories. For more discussion of the relation between the TDT and MTC, cf. our paper [8].

The MTC problem may be dealt with in many different ways. Due to a space limit we refer the reader to a number of approaches we proposed in our earlier papers.

3 The Proposed Approach

3.1 The Techniques Employed: HMM, LDA and the Logistic Regression

Hidden Markov Models (HMM) [12]**.** As we need a model of the sequence (case) of documents, we assume here that the case is a realization of a stochastic process with the Markov property and hidden states, i.e., is a Hidden Markov Model (Chain) (HMM), denoted by $\lambda = (A, B, \pi)$ and characterized by the following parameters:

1. the number of hidden states N_S; the states $S_i \in S = \{S_1, \ldots, S_{N_S}\}$, may be here interpreted as corresponding to the stages of the business process represented by a given case,
2. the number of distinct observation symbols $u_i \in U = \{u_1, \ldots, u_{N_U}\}$; here the observations are the whole documents and we discuss their representation in what follows – in the explanation of the line 5 of the algorithm shown in Fig. 1,
3. the state transition probability distribution, denoted as $A = [a_{ij}]_{1 \leq i,j \leq N_S}$, i.e., $a_{ij} = P(q_{t+1} = S_j \mid q_t = S_i)$, where q_t denotes the (hidden) state of the stochastic process at time t,
4. the observation symbols probability distribution $b_j \in B = \{b_1, \ldots, b_{N_S}\}$ defined for each state S_j, i.e.:

$$b_j : U \to [0, 1], \qquad b_j(u_i) = P(O_t = u_i \mid q_t = S_j) \tag{3}$$

   where O_t denotes an observation generated at time t; $O = (O_1, \ldots, O_T)$ will denote the whole sequence generated by the HMM which corresponds here to the sequence of documents (a case),
5. the initial probability distribution π over the state space S, i.e., $\pi(S_i) = P(q_1 = S_i)$.

For our purposes the original basic version of the HMM, as described above, seems to be not adequate. A possible extension [14] consists in adding covariates to condition the probabilities of the transitions and observations. We use a vector of covariates $cov_t = [cov_t^1, \ldots, cov_t^k]$ for the observation distribution conditioning which leads to the following modified form of (3):

$$b_j(u_i) = P(O_t = u_i \mid q_t = S_j, cov_t) \tag{4}$$

There are three basic problems related to the HMMs [12]:

– the evaluation problem, i.e., how to efficiently compute the probability of an observation sequence $O = (O_1, \ldots, O_T)$ given an HMM λ,
– the decoding problem, i.e., given an HMM λ and an observation sequence $O = (O_1, \ldots, O_T)$ what is a most probable (in some sense) sequence of states $(S_1, \ldots, S_T)$ which led to the generation of sequence O,

– the learning problem, i.e., given an HMM $\lambda = (A, B, \pi)$ and a sequence of observations O how to adjust λ's parameters A, B and π so as to maximize the probability of O, i.e., $P(O \mid \lambda)$.

In our algorithm we are dealing mostly with the first and third problem but the second problem is also of interest from the point of view of possible future modifications of our approach.

Thus, we may adopt an HMM λ_c as a rich generative model of sequences $\sigma =< d_1, \ldots, d_T >$ belonging to a given category c. We will discuss this in more detail later, including the form of the covariates involved, in the explanation of the line 5 of the algorithm shown in Fig. 1.

Latent Dirichlet Allocation (LDA) [4]**.** The Latent Dirichlet allocation (LDA) is a generative probabilistic model of a collection of documents (a corpus). Basically, it assumes that there is a set of k topics[1] $Z = \{z_j\}$ and each document $d \in D$ of the corpus deals with a mixture θ_d of them, i.e., $\theta_d : Z \rightarrow [0, 1]$ such that $\sum_j \theta_d(z_j) = 1$. Each topic z_j is, in turn, a distribution over a set of words (vocabulary) $V = \{w_i\}$, i.e., $z_j : V \rightarrow [0, 1]$ and $\sum_i z_j(w_i) = 1$.

It is assumed that for the whole corpus a parameter denoted by β is fixed and each topic distribution $z_j \in Z$ is sampled from the Dirichlet distribution with parameter β over the space of all multinomial (categorical) distributions over the vocabulary V. Another parameter set for the whole corpus is α which is the parameter of the Dirichlet distribution used to sample the mixtures of topics, to be explained below. Then, a document d, belonging to a corpus characterized by the values of parameters α and β, is assumed to be generated in the following process:

1. First, the length of the document in words, N, is sampled according to the Poisson distribution with the parameter ξ.
2. Second, the mixture of topics θ_d is sampled for the document according to the Dirichlet distribution with the parameter α.
3. Finally, for each of the N positions of words assumed to comprise the document d, first a topic z_j is sampled using the multinomial distribution θ_d and then a word $w \in V$ is chosen using the multinomial distribution related to the topic z_j.

Now, if we are given a corpus of documents we can observe only the values of the variables corresponding to the particular positions of the words within documents. All other random variables mentioned in the description of the generative process above are hidden. There exists a number of approaches to infer the posterior distributions of the hidden variables and to estimate parameters α and β [4]. Using one of them we obtain an LDA model of the corpus. Let us denote its part which will be useful for our further considerations as:

$$L = (\{z_j\}_{j=1,\ldots,k}, \{\theta_d\}_{d \in D}) = (Z, \Theta) \tag{5}$$

[1] To shorten the notation we will denote the topic in the same way as the distribution on the words defining it.

i.e., we have a set of k multinomial distributions z_j over the set of words V for all k topics and for each document $d \in D$ we have a mixture of topics θ_d characterizing it. We are also in a position to determine the representation of a new document $d^\star \notin D$ using the LDA model obtained.

3.2 The Algorithm

Here we assume that the incoming document $d^\star$ has been first properly classified to a category and the algorithm presented assigns a case to $d^\star$. We briefly discuss the question of category assignment in Sect. 3.3.

The general scheme of the proposed algorithm is presented in Fig. 1. Now we will discuss its particular lines, referring to the numbers shown in Fig. 1. In the next section we present the results of the computational experiments carried out using the R environment and its various packages, thus while describing here particular steps of the algorithm we will refer to its more general aspects as well as to the aspects specific for the assumed implementation.

Line 3. The document-term matrix forms a standard representation of the collection of documents in the vector space model [2]. The set of terms (the vocabulary) used to represent the documents is denoted as V. Here we employ the weights of the terms (keywords) in documents equal to the frequencies of their occurrence within those documents, i.e. the `tf` weighting scheme. This is the format preferred for the LDA analysis of the collection.

Line 4. An LDA model $L = (Z, \Theta)$ is constructed for the whole collection of training documents belonging to category c. The number of topics should be

1: **Initialization stage**
2: **for all** categories $c \in C$ **do**
3: *create* a document-term matrix
4: *create* an LDA model, LDA_c, for the collection $D_c \subseteq D$ of the documents belonging to category c
5: *train* an HMM model, λ_c, using all cases belonging to D_c
6: **end for**
7: **Classification stage**
8: $d^\star \leftarrow$ newly arrived document
9: $c^* \leftarrow$ category assigned to $d^\star$
10: *represent* $d^\star$ using the model LDA_c
11: **for all** ongoing cases σ_i **do**
12: *compute*, with respect to the HMM λ_c, the conditional probability of the case σ extended with the document $d^\star$, $< \sigma_i, d^\star >$, under the condition that the sequence σ has been generated, i.e. $P_{\lambda_c}(< \sigma_i, d^\star > | \, \sigma)$
13: **end for**
14: choose the case σ_i with the highest $P_{\lambda_c}(< \sigma_i, d^\star > | \, \sigma)$ and assign $d^\star$ to this case.

Fig. 1. A general scheme of the proposed algorithm

chosen experimentally but should not be too large as that number implies the number of parameters that have to be learned during the training of the HMM, in line 5.

Line 5. In this step, first, the representation of each document $d \in D$ provided by the obtained LDA model L in the form of a distribution θ_d is transformed into a binary vector[2], $d = [d_1, \ldots, d_k] \in \{0,1\}^k$, of dimension k in such a way that if the probability of a given topic z_j according to θ_d is greater than a threshold value τ (in the experiments $\tau = 1/k$), then $d_j = 1$ and otherwise $d_j = 0$, i.e.:

$$\theta_d \longrightarrow d : d_j = \begin{cases} 1 & \text{if } \theta_d(z_j) \geq \tau \\ 0 & \text{otherwise} \end{cases} \qquad j = 1, \ldots, k \tag{6}$$

Then, all cases present in collection D are used to train the HMM with a number of states N_S chosen experimentally and observations identified with the binary vectors d_j defined in (6). The observation probability distributions (3)–(4) are assumed to be multivariate Bernoulli distributions, i.e.,

$$b_j(u_i) = P(d \mid q_t = S_j) = \prod_{j=1}^{k} P(d_j = 1 \mid q_t = S_j)^{d_j} * P(d_j = 0 \mid q_t = S_j)^{1-d_j} \tag{7}$$

Actually, we are using a modified form of the formula (4) as we use the covariates for our observation distributions and the logistic regression to take them into account. Thus, in our case the following formula is employed:

$$\text{logit}(P(d_j \mid q_t = S_j)) = \omega_1 cov_t^j + \omega_0 \tag{8}$$

where the vector of covariates $cov = (cov_t^1, \ldots, cov_t^k)$ at time t is defined as follows:

$$cov_t^j = \theta_{d_{t-1}^{tfn}} \cdot z_j = \sum_{i=1}^{|V|} d_{t-1,i}^{tfn} * z_j^i \qquad j = 1, \ldots, k \tag{9}$$

where:

- $|V|$ denotes the size of the vocabulary,
- $d_{t-1}^{tfn} = (d_{t-1,1}^{tfn}, \ldots, d_{t-1,|V|}^{tfn})$ denotes the document occurring in the case at the preceding position (at time $t-1$ in the parlance of the HMM modeling) which is represented by its normalized version present in the document-term matrix created in line 3 of the algorithm shown in Fig. 1; the normalization takes the following form:

$$d_{t-1,i}^{tfn} = \frac{d_{t-1,i}^{tf}}{\max_j d_{t-1,j}^{tf}} \qquad i = 1, \ldots, |V| \tag{10}$$

  where $d_{t-1,i}^{tf}$ denotes the i-th coordinate of the vector representing the document in the document-term matrix before normalization,

[2] To simplify notation we denote this vector as d, i.e., in the same way as the document $d \in D$.

– z_j is the probability distribution representing the j-th topic, obtained as a part of the LDA model of the collection, which is here treated as a vector, i.e., $z_j = (z_j^1, \ldots, z_j^{|V|})$, $\sum_{i=1}^{|V|} z_j^i = 1$.

The usage of the covariates defined as above makes it possible to better model the patterns of the similarity/dissimilarity of the documents neighboring in a sequence belonging to a given category. More on that in the discussion provided in Sect. 3.3.

Line 8. A new document $d^\star$ to be classified is first represented both in terms of the document-term matrix mentioned in line 3 as well as in terms of the LDA model mentioned in line 4.

Line 9. As it is mentioned earlier, we assume that the document $d^\star$ is already classified to a category. In our previous work we usually use the k-nearest neighbors algorithm to do that. The current use of the LDA models opens new possibilities and in our further work we will check the efficiency of the method based on the LDA model.

Line 12. In order to select a case to which document $d^\star$ should be classified we compute for each on-going case $\sigma_i =< d_{i_1}, \ldots, d_{i_T} >$ and $d^\star$ the following index:

$$P(d^\star \mid \sigma_i, \lambda) = \frac{P(d_{i_1}, \ldots, d_{i_T}, d^\star|\lambda)}{P(d_{i_1}, \ldots, d_{i_T}|\lambda)} \tag{11}$$

which may be interpreted as the probability of the event that document $d^\star$ makes up the continuation of the case σ_i. In line 14 simply the case for which the probability (11) is highest is selected and the document $d^\star$ is assigned to it.

3.3 Discussion

The essence of the proposed algorithm, shown in Fig. 1, is relatively simple: the succession of the documents within cases is modeled using an HMM whose parameters are learned on the training data and a new document $d^\star$ is suggested to be added to a case for which it is the most probable successor (we do not consider here for simplicity the situation when a new case has to be established; for some solutions of this subproblem the reader is referred to our papers [8,19] as well as, e.g., to [15]). However, a few points do require some extra comments.

It should be noted that several representations of the documents are employed. The first is the standard vector space model based representation using the `tf` weighting scheme which is then employed to create an LDA model of the collection of documents[3]. The LDA based representation is then simplified,

[3] All text processing considered in this paper is carried out separately for each category $c \in C$, which will not be explicitly mentioned again, and, moreover, we will refer to the collection of documents having in mind its subset comprising documents belonging to one category.

namely it is turned into a binary representation, for the purposes of the HMM (see further discussion below). Finally, the original `tf` based representation is normalized/scaled for the purposes of the covariates computation.

The decision on the assumed documents representation is, of course, strongly connected with the form of the observation distributions used for the HMM based cases modeling. The first important assumption we adopted is that about independence of the features representing documents, i.e., terms/keywords in the standard vector space model representation or topics in case of the LDA. While this assumption is obviously incorrect in general, still it is usually assumed as otherwise the number of parameters of multivariate distributions makes effective and efficient learning practically impossible. Then, we have tried several options using both the Boolean representations of documents and their weighted forms, the former combined with the multinomial distribution and the latter combined with the Gaussian distribution. A multinomial distribution becomes cumbersome already for relatively small vocabularies V, requiring $N_S|V|$ parameters to be learned. In our experiments the vocabulary, already aggressively reduced, was composed of ca. 250 terms. The use of the LDA models makes it possible to reduce the number of features and at the same time provides for a more semantic rich representation. The number of parameters to be learned for the observation distributions is now equal $2N_S k$, where N_S is the number of states and k is the number of LDA topics. In our experiments the "binarized" version of the LDA representation proved to be most effective.

Actually, only after including covariates to a binary LDA representation via the logistic regression we have obtained satisfactory results in our experiments. The covariates are defined in such a way that the observation distribution – at a given point in time/position in the case – depends not only on the current state but also on the actual form of the preceding document expressed using normalized `tf` based representation. Formula (9) makes it possible to model the patterns of dependency between documents neighboring within a case such that occurrence in the preceding document of the terms strongly represented in a given LDA topic increases or decreases the probability of this topic in the next document in the sequence.

The proposed solution is based on a rather simple extension of the classic HMM. An interesting and natural alternative seems to be the use of a discriminative model, such as, e.g., the conditional random field. However, it should be noted that the MTC task resembles rather a time-series prediction problem than a sequential supervised learning problem [6]. In particular, in general, we do not assume the availability of the training data comprising cases where each document is assigned to a class (a label). Such a labeling may be envisaged, e.g., assuming that a specific stage of a business process may be associated with each document but this leads to a different class of possible approaches referring to the concept of business processes mining which we do not consider here. Anyway, in our research agenda for the MTC problem we consider the use of the Hidden Conditional Random Fields [10] which do not require labeled training sequences.

4 Computational Experiments

We have verified the proposed algorithm using an enlarged version of the collection of documents we adopted and used in our previous works. A detailed description of the collection may be found in [17,18]. The starting point is the set of articles on computational linguistics available in the framework of the ACL Anthology Reference Corpus (ACL ARC) [3]; see also http://atmykitchen.info/datasets/acl_rd_tec/cleansed_text/index_cleansed_text.htm. We use a subset of 664 papers which are composed of sections. In order to group the documents into categories we cluster the whole set of 664 papers into 6 clusters (the number 6 has been chosen experimentally to obtain reasonably sized categories). Then, we treat each paper as a case composed of documents corresponding to the sections of this paper.

Thus, we obtain 664 cases comprising 6884 documents in total. The number of cases and a cut-off point in each of them are randomly chosen. All documents at the cut-off positions are treated as test data while the documents following them are deleted from the collection. In each experiment, for each category a number of test documents has been selected proportionally to the size of this category, 64 documents in total in each experiment, i.e., 10 % of cases are each time treated as on-going.

The results obtained, averaged over 10 runs and 6 categories, are the following: microaveraged and macroaveraged accuracy of classification equal 0.54 and 0.57, respectively. The results are encouraging though one can well imagine a number of ways the proposed algorithm may be tuned and there seem to be a real potential for improvement thanks to employing a more semantic oriented document representation and an explicit modeling of dependencies between the documents within cases. In our previous papers we reported the results for other approaches we proposed earlier, including also a recent technique developed for the topic tracking task in TDT. However, most of them concerned a smaller subset of the ACL ARC corpus and also a smaller number of cases are there assumed to be on-going. It should be noted that if a case is considered as a class the respective classification problem gets usually more difficult with the growing number of classes; cf., e.g., [5]. However, recently we have tested (and compared against its newly proposed modified version) the method introduced in [18] on the same, larger version of the ACL ARC corpus which is adopted in this paper. We have obtained comparable results but the current proposed solution attempts to grasp the logic behind the order of the documents in a case in a more explicit way and is thus more promising as a starting point for some further improvements.

All computations are carried out using the R platform [11] and the following packages: `tm` [7], `topicmodels` [9], `depmixS4` [14] and our own R scripts. The most important parameters of the methods involved are the following: for the LDA – the number of topics $k = 30$, the α parameter of the Dirichlet distribution $= 1.67$, i.e., $50/k$, the beta parameter is automatically estimated; for the HMM – the number of states $= 6$, observation distributions are binomial (actually, Bernoulli as 1 trial is assumed) with the logit link.

5 Concluding Remarks

We have proposed a novel hybrid approach to solving the new multiaspect text categorization (MTC) problem proposed in our previous works. In comparison to our earlier approaches it assumes as a point of departure a more sophisticated explicit model of the whole collection of documents and, in particular, of the sequences of documents forming cases. In the new hybrid approach proposed, our earlier solution proposal based on the HMM is combined in a synergistic way with the LDA modeling of the collection of documents which certainly opens new vistas on the capability of this modeling. In particular, the possibility to link the probability of occurrence of an LDA topic in a given document with the vocabulary of the preceding document seems to be particularly interesting and promising. This is a type of dependency modeling we are looking for, i.e., such which to some extent abstracts from the actual value of the features of the documents and makes it possible to discover more universal patterns typical for different cases belonging to the same category.

Acknowledgments. This work is supported by the National Science Centre under contracts no. UMO-2011/01/B/ST6/06908 and UMO-2012/05/B/ST6/03068.

References

1. Allan, J. (ed.): Topic Detection and Tracking: Event-based Information. Kluwer Academic Publishers, Norwell (2002)
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. ACM Press and Addison Wesley, New York (1999)
3. 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), pp. 1755–1759. Marrakesh, Morocco (2008)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
5. Bayou, L., Espes, D., Cuppens-Boulahia, N., Cuppens, F.: Security issue of WirelessHART based SCADA systems. In: Lambrinoudakis, C., et al. (eds.) CRiSIS 2015. LNCS, vol. 9572, pp. 225–241. Springer, Heidelberg (2016). doi:10.1007/978-3-319-31811-0_14
6. Dietterich, T.G.: Machine learning for sequential data: a review. In: Caelli, T.M., Amin, A., Duin, R.P.W., Kamel, M.S., de Ridder, D. (eds.) SPR 2002 and SSPR 2002. LNCS, vol. 2396, pp. 15–30. Springer, Heidelberg (2002)
7. Feinerer, I., Hornik, K., Meyer, D.: Text mining infrastructure in R. J. Stat. Softw. **25**(5), 1–54 (2008)
8. Gajewski, M., Kacprzyk, J., Zadrożny, S.: Topic detection and tracking: a focused survey and a new variant. Informatyka Stosowana **2014**(1), 133–147 (2014)
9. Grün, B., Hornik, K.: topicmodels: An R package for fitting topic models. J. Stat. Softw. **40**(13), 1–30 (2011). http://www.jstatsoft.org/v40/i13/
10. Quattoni, A., Wang, S.B., Morency, L., Collins, M., Darrell, T.: Hidden conditional random fields. IEEE Trans. Pattern Anal. Mach. Intell. **29**(10), 1848–1852 (2007). http://dx.org/10.1109/TPAMI.2007.1124

11. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2014). http://www.R-project.org
12. Rabiner, L.: A tutorial on HMM and selected applications in speech recognition. Proc. IEEE **77**(2), 257–286 (1989)
13. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. **34**(1), 1–47 (2002)
14. Visser, I., Speekenbrink, M.: depmixS4: An R package for Hidden Markov Models. J. Stat. Softw. **36**(7), 1–21 (2010)
15. 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, pp. 688–693. ACM, New York (2002)
16. Zadrożny, S., Kacprzyk, J., Gajewski, M., Wysocki, M.: A novel text classification problem and its solution. Tech. Trans. **4–AC**, 7–16 (2013)
17. Zadrożny, S., Kacprzyk, J., Gajewski, M.: A new two-stage approach to the multiaspect text categorization. In: 2015 IEEE Symposium on Computational Intelligence for Human-like Intelligence, CIHLI 2015, Cape Town, South Africa, December 8–10, 2015, pp. 1484–1490. IEEE (2015)
18. 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, 17–19 August 2015 (2015)
19. Zadrożny, S., Kacprzyk, J., Gajewski, M.: On the detection of new cases in multiaspect text categorization: a comparison of approaches. In: Proceedings of the Congress on Information Technology, Computational and Experimental Physics, pp. 213–218. AGH University of Science and Technology (2015)
20. Zaki, M.J.: SPADE: an efficient algorithm for mining frequent sequences. Mach. Learn. **42**(1/2), 31–60 (2001)