INFORMATICA, 2010, Vol. 21, No. 4, 471–486

Statistical Classification of Scientific Publications

Vaidas BALYS, Rimantas RUDZKIS
Vilnius University Institute of Mathematics and Informatics
Akademijos 4, LT-08663 Vilnius, Lithuania
e-mail: rudzkis@ktl.mii.lt

Received: July 2009; accepted: September 2010

Abstract. The problem of automatic classification of scientific texts is considered. Methods based on statistical analysis of probabilistic distributions of scientific terms in texts are discussed. The procedures for selecting the most informative terms and the method of making use of auxiliary information related to the terms positions are presented. The results of experimental evaluation of proposed algorithms and procedures over real-world data are reported.

Keywords: statistical classification, probabilistic distribution, parametric estimation, auxiliary information, informative terms.

1. Introduction

The automatic classification of a chunk of scientific text is a topical task due to the rapid growth of volumes of published material as well as the ongoing shift from a paper based publishing to e-publishing. Scientific information needs to be properly processed and enhanced so that it could be conveniently found, retrieved, and reused whenever needed. Automatic classification is one of processing activities that supplements a raw text with an additional higher-level information which ensues an efficient navigation within the vast space of scientific material.

There is a long list of plain text classification algorithms that proved to be highly efficient and popular; see, e.g., Sebastiani (2002) for an overview from a machine learning point of view. Such algorithms are instrumental in solving natural language texts processing problems. These problems of applied nature are high on the research agenda; see Marinčič *et al.* (2009) and Maučec *et al.* (2009) for a couple of interesting recent studies. Nevertheless, scientific texts differ from the natural language texts to a certain extent, and there is a need for studies addressing the specificity of scientific contents.

In paper Rudzkis (2006) we presented a novel approach to the classification of scientific texts based on the statistical analysis of scientific term distributions over texts. The proposed methods mathematically formalise and extend the heuristic idea of contextual evidence called "identification cloud"; see, e.g., Hazewinkel (2004). The early results of experimental evaluation of the proposed methods were presented in Rudzkis and Balys (2008).

In this paper, we propose a number of refinements and improvements to the methods. These include a procedure for selecting the most informative scientific terms as well as

a method to make use of auxiliary information related to the location of terms in the text. The experiments on a real publications database were conducted and the results confirming the positive influence of these improvements are reported.

The paper is organised as follows. In Section 2, the outlines of the proposed approach to the classification of scientific texts is presented with Section 2.4 devoted to the refined procedure for selecting the most informative terms. In Section 3, a new method of including the auxiliary information into the models and algorithms is proposed. Section 4 is devoted to the results of experimental evaluation. And the last section is for the conclusions.

2. The Model

Scientific papers usually have certain pieces of meta-data (usually assigned by authors) describing the topics and the main ideas of the contents. MSC classifiers and keywords are the most common examples within the mathematical society. MSC classifiers are used for the classification by definition while keywords can serve a number of different purposes. However, having a controlled vocabulary of keywords fixed one can treat the assignment of keywords as the classification of a paper. Keywords and certain groups of them describe classes of related documents. In this case, assignment of keywords is effectively the same activity as assignment of MSC classifiers and both of them may be completed by using statistical classification algorithms. In this paper, the problem of automatic classification of scientific publications is considered keeping in mind that the real applied problem of interest is the assignment of either MSC classifiers or keywords from controlled vocabulary.

Further in this section, a brief introduction to the proposed approach to the classification based on the statistical analysis of term distributions over texts is presented. For a comprehensive review refer to Rudzkis *et al.* (2006).

2.1. *Definitions and Notation*

Let us introduce some definitions and notation that allow us to reduce the applied problem of classification of scientific papers to the common problem of statistical classification of multidimensional data.

Let K denote some classification system of scientific texts which is identified with a set of all possible labels of the classes in that system. Let V be a vocabulary (set) of scientific terms of a certain field that are relevant to the classification of texts. The chronologically enumerated vector of the article a elements $(a_1, \ldots, a_d)$, $d = d(a)$, where $a_i \in V$ and not necessarily $a_i \neq a_j$, is called the projection of the article a. For the convenience, we identify the projection of an article a with an infinite sequence $(a_1, a_2, \ldots)$, where $a_i = 0$ for all $i > d(a)$. Here $0 \in V$ denotes an additional zero term which does not exist in reality. Let A be a set of projections of all the articles from a certain scientific field. In what follows the word "projection" is omitted and $(a_1, a_2, \ldots)$ is called just an article.

Let N be a set of natural numbers. Let an article $a \in A$ and a set of indices $I \subset N$ be chosen randomly. The article is attributed to the class $\eta = w_\nu(a)$ in the system K. A problem of classification is to determine the unknown class η using the observed vector $a_I = (a_i, i \in I)$.

2.2. *Probability Distributions*

Since (a, I, η) is the result of a random experiment, the probability distribution in the set K is defined by

$$Q(w) = \mathbb{P}\{\eta = w\}, \quad w \in K. \tag{1}$$

Let Y be a set of all possible values of a_I. In the set Y the following conditional probability distributions are defined:

$$P(y) = \mathbb{P}\{a_I = y \mid |I| = d(y)\},$$

$$P(y|w) = \mathbb{P}\{a_I = y \mid |I| = d(y), \eta = w\}, \quad w \in K, \tag{2}$$

where $d(y) = \dim(y)$ is the dimension of vector y and $|I| = \text{card}(I)$ is the cardinality of set I.

If η and $|I|$ are independent, after observing a_I, the posterior probability of the random event $\{\eta = w\}$ is determined by $Q(w|a_I) = Q(w) \cdot \psi_w(a_I)$, where

$$\psi_w(y) = P(y|w)/P(y), \quad y \in Y. \tag{3}$$

The functional ψ_w reflects how the probability to observe a text changes if this text appears to be classified under certain class.

Using the distributions, introduced in (1) and (2), the Bayes classifier can be defined. In case when the loss function is trivial, it is determined by the following equation:

$$\widehat{\eta} = \arg\max_{w \in K} P(a_I|w)Q(w),$$

in which $\psi_{(\cdot)}(a_I)$ can be substituted for $P(a_I|\cdot)$:

$$\widehat{\eta} = \arg\max_{w \in K} \psi_w(a_I)Q(w). \tag{4}$$

2.3. *Inference*

In order to use classification method (4), the distribution Q and the functional ψ must be estimated.

Let us have the learning sample of the observed parts of n texts and their classification results $X = (y(1), \eta(1)), \ldots, (y(n), \eta(n))$, where $\eta(i) \in K,\ y(i) \in Y, Y = \{y = (y_1, \ldots, y_d)\colon y_i \in V,\ d \in N\}$.

$\widehat{Q}(w) = \sum_{j=1}^{n} 1_{\{\eta(j)=w\}}/n$ is the empirical analogue of $Q(w)$. For the estimation of functional ψ additional definitions are needed. Let the index $\tau \in I$ be a random variable. The distribution on set V is defined by $P(v) = \mathbb{P}\{a_\tau = v\}$ and the corresponding conditional distribution is given by $P(v|w) = \mathbb{P}\{a_\tau = v|\eta = w\}$, $w \in K$. The following assumption substantially simplifies the procedures of estimation.

ASSUMPTION (Conditional stationarity and independence). *Let for all $y \in Y$ and $w \in K$ hold*

$$P(y|w) = \prod_{i=1}^{d} P(y_i|w),$$

where $d = d(y)$ as before is the dimension of the vector y.

Now the definition (3) can be changed to

$$\psi_w(v) = P(v|w)/P(v), \quad v \in V,\ w \in K, \tag{5}$$

and the Bayes classification rule for classifying the article is determined by

$$\widehat{\eta} = \arg\max_{w \in K} \left[Q(w) \prod_{i \in I} \psi_w(a_i) \right]. \tag{6}$$

The assumption and the definition (5) ignore information that can be derived from the order of the terms in the text. Therefore, in Rudzkis *et al.* (2006) we introduce a weaker assumption that substitutes independence with Markov property. However, experiments did not show any advantages of this approach. The model becomes more complicated which yields higher error rate of statistical identification. Nevertheless, this more adequate model could be used if much bigger learning samples are available. In this paper, we limit ourselves to the simpler case of the independence assumption.

In order to use (6), the functional $\psi_w(v)$ still has to be estimated which is easier than estimating $\psi_w(y)$ used in (4). In Rudzkis *et al.* (2006) we proposed a procedure for that which consists of three steps (and possibly the fourth – parametrisation). Firstly, the empirical estimates of the probabilities $P(\cdot)$ and $P(\cdot,\cdot)$ are calculated by counting the appropriate frequencies and these estimates are used in (5) yielding empirical $\widetilde{\psi}_w(v)$. Secondly, the smoothing is performed: the unreliable estimates, i.e., the ones based on too few observations, are modified. The next step is selecting the most informative terms. This step is covered in detail in the following section.

2.4. *The Informative Terms*

When deciding if article should be (or should not be) attributed to a certain class w only a part of the terms (denoted $L(w)$) that are the most informative towards that class is actually used. The basic approach to the selection of this subset together with a constructive

procedure were presented in Rudzkis *et al.* (2006). Here, we present some new methods and procedures. These procedures were experimentally compared and the results of the comparison are reported in the section 4.

Let $h = |V|$. The functional $\psi_w(\cdot)$ determines the arrangements of set V for each $w \in K$:

$$\psi_w(v_1) \geqslant \psi_w(v_2) \geqslant \cdots \geqslant \psi_w(v_h), \quad v_{(\cdot)} \in V. \tag{7}$$

Firstly, we arrange the set V as (7) by using estimate $\widetilde{\psi}_w(\cdot)$ instead of unknown $\psi_w(\cdot)$. The set $L = L(w) \subset \{1, \ldots, h\}$ consists of indices of the terms that have $\widetilde{\psi}_w(v_k)$ significantly differing from 1.

The straightforward method for selecting L is to pick a fixed number of terms (this number may come from prior knowledge or may be derived by conducting experiments and choosing the optimal value) that corresponds to the biggest and analogously another number that corresponds to the smallest values of $\widetilde{\psi}_w(\cdot)$. In this case,

$$\overline{L} = \{1, \ldots, s\}, \qquad \underline{L} = \{h - l, \ldots, h\}, \tag{8}$$

where s and l – some constants (the same for all the $w \in K$). Obviously, $L = \overline{L} \cup \underline{L}$.

Let us now consider more adequate adaptive methods that choose the number of the most informative terms for each class separately. Let us consider the hypothesis

$$H_0\colon \psi_w(v) = 1$$

with an alternative

$$H_1\colon \psi_w(v) > 1,$$

and let $\overline{\alpha}(v)$ denote a p-value. Analogously, let us consider the same hypothesis, but with another alternative

$$H_1\colon \psi_w(v) < 1,$$

and let $\underline{\alpha}(v)$ denote the corresponding p-value.

Choosing the level of significance α, the set of the most informative terms may be defined by equations

$$\overline{L} = \{k\colon \widetilde{\psi}_w(v_k) > 1,\ \overline{\alpha}(v_k) < \alpha\}, \qquad \underline{L} = \{k\colon \widetilde{\psi}_w(v_k) < 1,\ \underline{\alpha}(v_k) < \alpha\}. \tag{9}$$

There are a couple of alternative methods for choosing the most informative terms. We present them in the experimental part of this paper. In paper Rudzkis *et al.* (2006), relations between $\underline{\alpha}(v)$ and α as well as $\overline{\alpha}(v)$ and α are presented that render the proposed procedures practical.

3. Auxiliary Information

By selecting only a part of scientific terms from the text and using stationary distributions $P(\cdot)$ and $P(\cdot|\cdot)$ we ignore the context between the terms as well as the location of these terms in the text. When encountering long and non-homogeneous texts which is the case with scientific publications, it is natural to consider at least a part of this information as it is obvious that a term has different discriminative weight depending on which part of an article text it is observed in.

For example, assume that we observe some term twice in the abstract and the introduction of the document classified under some class w_1. On the other hand, we observe the same term three times in the fifteenth page and the proofs sections of the document classified under another class w_2. Basing only on these observations, one would naturally conclude that this term provides stronger evidence (and should get the higher discriminative weight) towards the class w_1 as opposed to the class w_2 even though the frequencies of observations support the opposite proposition. This reasoning is based on the intuitive understanding of what parts of the text are more informative towards the topics of the paper.

Let us now assume another situation. We obtain a new paper to be classified and in the introduction we observe some term which is highly informative towards some class w_1. We also observe another term in the fifteenth page that is equally informative towards another class w_2. One would safely conclude that paper should be attributed to the class w_1 despite the quantitative equality of discriminative evidence towards both classes. Here, we follow the same reasoning: highly informative terms in some distant parts of the paper may be not that strong evidence as some moderately informative terms in the most important parts of the paper.

Further in this section, we propose a way to mathematically define such heuristic reasoning about the influence of the location of terms in the text and then to use it in the proposed algorithms.

Let us redefine the projection of an article by $a = ((a_1, \lambda_1), \ldots, (a_d, \lambda_d))$, $d = d(a)$, where λ_i is a scalar (in a more general setting it could also be a vector) representing auxiliary information for a term a_i related to the location of this term in the text. There is an open question how this auxiliary information should be defined. Some measure of a distance from the beginning of the article is an obvious choice. A sequential number of term, word, sentence and paragraph are used for measuring such distance in this paper.

The logical or structural part of the article (abstract, main results, proof) is another strong option. One could also provide some vocabulary of words or phrases (not scientific terms) that mark the locations in the text where the most important things are usually presented (e.g., “this paper considers”, “the main idea is”, etc.) and then the distance from the closest of these terms may be measured. Neither of these options were used in the research because structural elements of the articles were unidentifiable in the data we possessed for the experiments. Similarly, we did not have a dictionary of the aforementioned phrases. Note, that these shortcomings may be overcome by developing and implementing automatic algorithms for identification of the structural parts of the paper as well as for collecting these important language phrases.

Let us have functional $\sigma = \sigma(\lambda_{(\cdot)})$ taking values in $[0, 1]$ which defines the weight of term depending on its location. Having learning sample $X_n = (y(1), \lambda(1), \eta(1)), \ldots, (y(n), \lambda(n), \eta(n))$, $\eta(i) \in K$, $y(i) \in Y$, enriched with auxiliary information $\lambda(j) = (\lambda_1(j), \ldots, \lambda_{d(j)}(j))$, $d(j) = d(y(j))$, we can derive the weighted estimates for $P(\cdot)$ and $P(\cdot|\cdot)$ as follows:

$$S(j) = \sum_{k=1}^{d(j)} \sigma(\lambda_k(j)), \tag{10}$$

$$\widetilde{P}^*(v) = \sum_{j=1}^{n} \sum_{k=1}^{d(j)} \sigma(\lambda_k(j)) \cdot 1_{\{y_k(j)=v\}} \Big/ \sum_{j=1}^{n} S(j), \tag{11}$$

$$\widetilde{P}^*(v|w) = \sum_{j=1}^{n} \sum_{k=1}^{d(j)} \sigma(\lambda_k(j)) 1_{\{y_k(j)=v, \eta(j)=w\}} \Big/ \sum_{j=1}^{n} 1_{\{\eta(j)=w\}} \cdot S(j). \tag{12}$$

These estimates are substituted in (5) for the unknown true values and then the same procedures for smoothing and selecting the most informative terms are conducted with this new estimate of $\psi_w(\cdot)$.

Analogously, the classification procedure (6) can be redefined as follows to take into account the positions of terms in the text to be classified:

$$\widehat{\eta} = \arg\max_{w \in K} \left[Q(w) \prod_{i \in I} \psi_w^{\sigma(\lambda_i(a))}(a_i) \right]. \tag{13}$$

In order to use the enhanced estimation procedures and classification method one has to provide the functional $\sigma(\lambda) = \sigma_\theta(\lambda)$ that describes the weight of the scientific term depending on its position in the text of the article. It is natural to assume that this weight should decrease with increasing distance from the beginning of the text. It could possibly become equal to zero at some distance thus limiting the length of texts to be analysed as well as to prevent problems of irrelevant terms in such distant parts of articles as proofs section.

We propose to use a piecewise linear function that can be defined by using recursive formula

$$\sigma_\theta^*(i) = \begin{cases} 1, & i = 0, \\ 0, & i > l_k, \\ \max\{0, \sigma_\theta^*(i-1) + \alpha_j\}, & i \in (l_{j-1}, l_j], \end{cases}$$

with parameter vector

$$\theta = (\alpha_1, l_1, \alpha_2, l_2, \ldots, \alpha_k, l_k).$$

Here, $0 = l_0 \leqslant l_1 < l_2 < \cdots < l_k$, $\alpha_{(\cdot)} \leqslant 0$ and $\alpha_i \neq \alpha_{i+1}$.

Then for a chosen parameter θ we define

$$\sigma(\lambda) = \sigma^*_\theta(\lambda).$$

4. Experimental Evaluation

4.1. *The Data*

The experiments were conducted on basis of nearly 15,000 articles from the field of probability theory and mathematical statistics kindly provided by the Institute of Mathematical Statistics, USA. 44 MSC classifiers (24 from 60XXX branch and 20 from 62XXX branch) were chosen for the experiments each having a learning set of at least 100 articles, resulting in a total of 5338 articles.

The dictionary of scientific terms was constructed by extracting all the keywords from the articles in the database. The single words that build up keywords-phrases were also added. This resulted in a list of 17,632 unique terms.

Three major parts of texts were available for each article: title, abstract, and full text (or simply text). This means that various learning/testing sets configurations were available. These configurations are referred to in format *text/abstract* where the first item denotes the parts of texts used for the learning documents and the second item denotes the parts of texts used for the testing documents.

4.2. *The Algorithms*

Let us recall the random experiment from Section 2.1 which results in random article $a \in A$, it's unobserved class η, and the observed vector of terms a_I.

For non-probabilistic algorithms (see *kNN*, *SVM*, and *LLSF* right below) a different representation of article's terms is used. Having fixed the order of all h terms of vocabulary V, the observed part of article is represented by an h-length vector in which the ith element stands for the weight of the ith term (from V) in this observed text. The binary weight scheme (1 if term is present in the observed part of the article, 0 if not) was used in this research.

Let us once again have the learning sample of the observed parts of n texts and their classification results $X = (y(1), \eta(1)), \ldots, (y(n), \eta(n))$ (see Section 2.3). As noted in the previous paragraph, for the non-probabilistic algorithms, $y(\cdot)$ is an h-length vector of zeroes and ones.

The algorithms:

- *IDC*. Algorithm based on the proposed approach.
- *nB (naive Bayes with additive smoothing)*; see Mitchell (1996). A rather similar probabilistic method that builds on the same assumption of conditional indepen-

dence and stationarity of terms' distributions over text. The equivalent of (6) classification rule is used:

$$\widehat{\eta} = \arg\max_{w \in K} \left[Q(w) \prod_{i \in I} P(a_i|w) \right].$$

$Q(\cdot)$ and $P(\cdot|\cdot)$ are empirically estimated from the learning set. When estimating $P(\cdot|\cdot)$ a fixed small number is added to the counts to eliminate estimates equal to zero (so-called additive smoothing).

- *kNN (k nearest neighbours)*; see Yang (1994). A common instance-based algorithm that defers the learning phase until the document to be classified arrives. Then a number of nearest documents from the learning set is chosen. The classification decision is derived from the classification decisions of these nearest documents (majority voting, distance-weighted voting, etc.). As documents are represented by a fixed-length binary vectors, a number of common distance measures may be used to find the nearest neighbours. In this paper, cosine of the angle between vectors was chosen as distance measure (higher accuracy of the algorithm was observed as compared to using Euclidian distance).
- *SVM (Support Vector Machines)*; see Vapnik (1995). Algorithm interprets documents as points in the h-dimensional Euclidian space and for each class calculates the hyperplane separating positive and negative examples from the learning set. Of all the possible separating hyperplanes the one that separates positive and negative examples by the widest margin is chosen. The extended methods are available for the cases when the examples cannot be cleanly separated. In the classification phase each possible class is assigned a probability depending on which side of the hyperplane the document falls as well as on the distance from it. The specific implementation SVM^{light} (Joachims) was used in this research.
- *LLSF (Linear Least Squares Fit)*; see Yang and Chute (1992). Let us similarly as with V fix the order of classes from K. Then a classification decision for a document may be represented as a fixed-length vector having exactly one element "1" while the remaining are zeroes. Algorithm assumes the existence of linear dependence between vector (consisting of zeroes and ones) representing terms and vector representing classification decisions of an article. The least squares fit method over learning data is used to find the matrix that defines this linear dependence. Then the classification phase is reduced to a simple multiplication of vector and matrix and finding the biggest element of a resulting vector.

All the considered algorithms (except for *kNN*) perform by analysing the positive and the negative examples of classes and building discriminative rules so that they classify learning data as correct as possible.

The algorithms implement the ranking procedure: for each document a list of classes that document could be attributed to with the corresponding weights (that roughly represent the probability that the document should be attributed to the class) is delivered. Then, depending on some threshold strategy, a subset of classes with the highest weights is chosen.

4.3. *Performance Measures*

The k-fold cross validation procedure with $k = 5$ was used to evaluate the algorithms. The common measures *precision* (Pr), *recall* (Re), and the harmonic average $F_1 = 2Pr \cdot Re/(Pr + Re)$; see, e.g., Yang (1999) were used to compare the true (denoted by K_T) and the guessed (denoted by K_A) classification decisions (sets of assigned class labels). All the above listed measures depend on the number of classes chosen by algorithm.

A measure similar to 11-point average precision (Yang, 1999) was used to estimate the efficiency without fixing the number of classes the algorithm selects for the document. Let us have a document that has to be classified. Assume that it has m classes (labels) assigned ($|K_T| = m$). The classification algorithm ranks all the available classes in decreasing order of relevance to the document (each algorithm has its own interpretation of relevance). Let us pick the highest ranked class from this list and assign its label to the document. Then pick the second highest ranked class and so on. Thus we gradually construct K_A set. Each time we assign a label we check if it coincides with any of truly assigned ones from K_T. When all the true labels from K_T are matched we halt the process. At this moment, value of recall is 1. Each time the newly assigned label coincided with one from the K_T, value of recall increased by $1/m$. At these points the precision was equal to a number of already matched classes divided by a number of steps taken. The average precision for a document is defined as the average of precision values at recall increase points: $Pr_{avg}(a) = (1/n_1 + 2/n_2 + \cdots + m/n_m)/m$ where a is article considered and n_i is the number of steps taken until i true labels were correctly assigned. The average over all documents from a testing set gives the measure of algorithm efficiency, denoted by Pr_{avg} and called *average precision*.

4.4. *Results*

In (9) we presented a statistical hypothesis testing based method for selecting the most informative terms. We will refer to this method as *hyp*. Another one, *hyp/stop* is defined by (8) and the values of s and l are now determined as

$$s = \max\{k\colon \overline{\alpha}(v_j) < \alpha,\ j = \overline{1,k}\},$$
$$l = \max\{k\colon \underline{\alpha}(v_{h-j}) < \alpha,\ j = \overline{0,k}\}.$$

One more method *hyp/fixed* is defined by yet the same (8), with the values of s and l determined as

$$s = |\{k\colon \widetilde{\psi}_w(v_k) > 1,\ \overline{\alpha}(v_k) < \alpha\}|,$$
$$l = |\{k\colon \widetilde{\psi}_w(v_k) < 1,\ \underline{\alpha}(v_k) < \alpha\}|.$$

This one is quite similar to *hyp* method as both of them choose the same number of informative terms. However, the terms may be different.

In Fig. 1, the results of the comparison of these most informative terms selecting procedures under *text/text* setting are presented. The F_1 measure was calculated for a

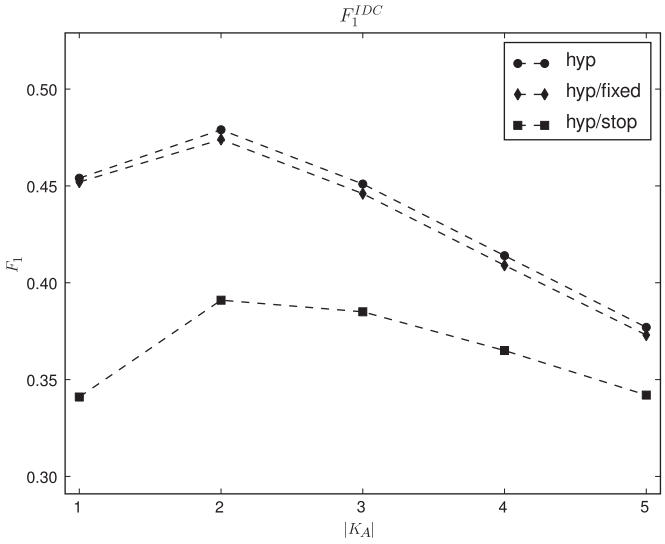

Fig. 1. The influence of the most informative terms selecting procedure on the performance.

number of fixed-number class assignment strategies (number varying from 1 to 5). Values of α from 0.001 to 0.01 were found to be optimal (the figure is presented for a case of $\alpha = 0.01$). It is evident that *hyp/stop* method performs the worst and *hyp* method – the best. The results when using *hyp/fixed* method are near to optimal. This can be explained as follows. When using long (full) texts for learning the probabilities are estimated rather reliably. In such a case it is improbable that very big or very small value of the $\psi_w(\cdot)$ estimate proves to be insignificantly different from 1. This means that *hyp/fixed* method which selects the terms having the biggest and the smallest values of this estimate makes almost the same decisions as *hyp* method. However, me must note that *hyp/fixed* method gets the number of terms to be chosen from *hyp* method and these numbers are different for each class and they also depend on the value of α. The non-adaptive method which selects the same numbers for all the classes performs much worse (comparably to *hyp/stop* method).

The Table 1 presents the estimated efficiency of algorithms (measured by average precision P_{avg}) over various combinations of learning and testing sets.

DF (document frequency) method was used to exclude the uninformative terms – all the terms that were observed in less than some fixed number of different documents were effectively removed from vocabulary – as according to Yang and Pederse (1997) it is among the best for non-aggressive feature space dimensionality reduction. The optimal number of different documents for each term was found to be between 3 and 6 – the size of vocabulary was reduced by a factor of 2–3 while efficiency of algorithms increased.

kNN algorithm (k ranging from 25 to 50) performed the worst while *SVM* came out as a winner by a slight margin. The efficiency of the *IDC* algorithm is very similar to that of the *SVM*. Comparing these results to the ones published in Rudzkis and Balys (2008) it is evident that *IDC* algorithm gained a substantial increase in efficiency because

Table 1

P_{avg} of algorithms for various learn/test sets configurations

	nB	*kNN*	*SVM*	*LLSF*	*IDC*
Title/title	0.512	0.485	0.503	0.500	0.502
Abstract/abstract	0.584	0.540	0.590	0.580	0.592
Abstract/text	0.598	0.562	0.602	0.594	0.604
Text/abstract	0.618	0.568	0.629	0.555	0.617
Text/text	0.653	0.586	0.661	0.628	0.658

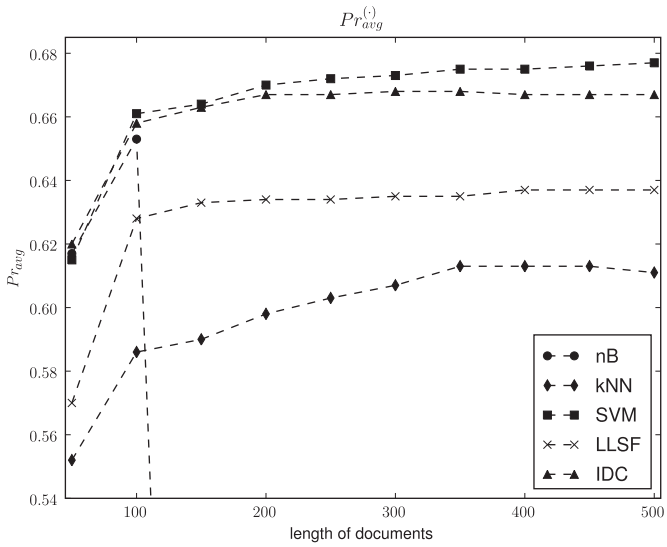

Fig. 2. The influence of text length on the performance.

the informative terms selecting procedure was changed from a *hyp/stop* variant to *hyp*. Another difference is that now not only the terms with weights bigger than 1 but also the terms with weights smaller than 1 are used which was not possible previously. Note, that the presented results are for the case when only the first 110 terms are used instead of full documents (in *text/text* setting) and *IDC* algorithm does not use auxiliary information.

The Fig. 2 shows how the average precision P_{avg} changes when longer texts (the length of text is measured in a number of scientific terms) are used for both learning and testing. The first 100–110 terms of text are the most useful as the average precision steeply increases. Starting from approximately 120 terms the performance of *nB* drastically decreases possibly due to the problem of irrelevant and unseen terms starting to show after the introductory parts of the articles are finished. The results for all the remaining algorithms almost stabilise when using texts that are longer than 300 terms while for the *IDC* there are weak signs of decrease.

The Table 2 presents the results obtained when *IDC* algorithm uses auxiliary information. The first four columns are for the Pr_{avg} results of the *IDC* algorithm and the last one – for the corresponding Pr_{avg} value of the *SVM* algorithm. The first column is for the case when auxiliary information is used only for the trimming of texts (only that part of the texts is used that has non-zero σ value, but the weights of all these terms are set equal to 1). The second column is for the case when auxiliary information is used only in learning, i.e., functionals $\psi_w(\cdot)$ are estimated by using (10)–(12) but the classification procedure is (6) that does not use weights. The third column is for the case when auxiliary information is used only in testing phase, i.e., classification procedure is (13) but the estimation of $\psi_w(\cdot)$ is performed by not using weights (i.e., all weights are equal to 1).

Table 2

The influence of extending *IDC* to use auxiliary information to the efficiency.

Pr_{avg} value when auxiliary information is used for ...				The corresponding
Trimming	Learning	Testing	Learning and testing	Pr_{avg} value for *SVM*
$\theta = (-0.005, 110)$				
0.658	0.658	0.659	0.659	0.661
$\theta = (0, 60, -0.01, 110)$				
0.658	0.659	0.659	0.660	0.661
$\theta = (0, 110, -0.5, 111, 0, 200)$				
0.667	0.668	0.669	0.671	0.670
$\theta = (0, 110, -0.5, 111, 0, 300)$				
0.668	0.668	0.671	0.673	0.673
$\theta = (0, 110, -0.5, 111, 0, 200, -0.2, 201, 0, 300)$				
0.668	0.669	0.672	0.675	0.673
$\theta = (0, 110, -0.5, 111, 0, 400)$				
0.667	0.668	0.671	0.673	0.675
$\theta = (0, 110, -0.5, 111, 0, 200, -0.2, 201, 0, 400)$				
0.668	0.670	0.673	0.676	0.675
$\theta = (0, 110, -0.5, 111, 0, 200, -0.3, 201, 0, 400)$				
0.668	0.670	0.674	0.678	0.675
$\theta = (0, 110, -0.5, 111, 0, 500)$				
0.667	0.669	0.673	0.676	0.677
$\theta = (0, 110, -0.5, 111, 0, 200, -0.2, 201, 0, 500)$				
0.667	0.670	0.675	0.679	0.677
$\theta = (0, 110, -0.5, 111, 0, 200, -0.2, 201, 0, 350, -0.2, 351, 0, 500)$				
0.667	0.670	0.677	0.681	0.677

The fourth column is for the case when auxiliary information is used both in learning and testing phases. The values of Pr_{avg} for the *SVM* algorithm in the fifth column are calculated for the trimmed texts similarly as in the first column for the *IDC* algorithm.

The weight functional σ is defined by considering one of the sequential numbers (term, word, sentence, paragraph) to be a descriptor of position. The experiments showed that the results are rather similar for all these cases. The one difference is that it is easier to choose the optimal value for the parameters vector θ in case of using paragraph or sentence number as the position descriptor. The results in Table 2 are presented in case when functional σ is defined by considering term number as a position descriptor.

It is evident from the results that by using auxiliary information the results can be improved for the *IDC* algorithm. The biggest effect is achieved when auxiliary information is used both for learning and testing. The best results are achieved using simple θ values that define a small number (2–3) of positions at which the weight is reduced by discrete steps of 20–30% until it is equal to zero. The more complicated models with the weight linearly decreasing over some intervals did not yield any improvement over this simple case.

5. Discussion and Conclusions

In this paper, the scientific texts classification methodology, based on the analysis of terminology distribution over text, is further developed.

New procedures for the selection of the most informative terms are formulated. Experimental evaluation confirms a significant increase in classification accuracy as compared to the results reported in the publication Rudzkis and Balys (2008).

The procedures of making use of auxiliary contextual information for the classification are presented. These procedures formalise the heuristic reasoning of different discriminative weight depending on the term location. By using the procedures, the classification accuracy is improved for very long texts.

Analogously, contextual information may well be used in more complicated cases when not only single terms but also pairs of them are considered. In such a case, a limited approach to analyse only neighbouring pairs may be generalised. Distant pairs (with some terms in between them) could be considered. And the pairs could be treated differently depending on the context between them. Preliminary experiments look encouraging but for a full-scale evaluation bigger databases are needed.

The classification algorithm, based on the proposed approach of statistical analysis of terminology distributions, not only slightly outperforms the analysed alternative ones, but also has some practical advantages. The relations of model parameters (e.g., term weights) are both explicit and interpretable. These relations gives valuable information about the considered scientific field. On the other hand, methods improvement by including some expert knowledge about the field is also possible. And the parametrisation, presented in Rudzkis *et al.* (2006), greatly simplifies this inclusion as the weights of terms can be calculated knowing only their order of importance towards some class.

References

Hazewinkel, M. (2004). Dynamic stochastic models for indexes and thesauri, identification clouds, and information retrieval and storage. In: Baeza-Yates, R. *et al.* (Eds.), *Recent Advances in Applied Probability*, pp. 181–204.

Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. In: *ECML '98 Proceedings of the 10th European Conference on Machine Learning*. Springer, Berlin, pp. 137–142.

Joachims, T. SVM^{light} implementation of support vector machines. `http://svmlight.joachims.org/`.

Marinčič, D., Tušar, T., Gams, M., Šef, T. (2009). Analysis of automatic stress assignment in Slovene. *Informatica*, 20(1), 35–50.

Mau–ìec, M.S., Brest, J. (2010). Reduction of morpho-syntactic features in statistical machine translation of highly inflective language. *Informatica*, 21(1), 95–116.

Mitchell, T.M. (1996). *Machine Learning*. McGraw-Hill, New York.

Rudzkis, R., Balys, V. (2008). Classification of publications based on statistical analysis of scientific terms distributions. *Aust. J. Stat.*, 37(1), 109–118.

Rudzkis, R., Balys, V., Hazewinkel, M. (2006). Stochastic modelling of scientific terms distribution in publications. In: J.M. Borwin and W.M. Farmer (Eds.), *Mathematical Knowledge Management*, *Lecture Notes in Computer Science*, Vol. 4108. Springer, Berlin/Heidelberg, pp. 152–164.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1), 1–47.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer, New York.

Yang, Y. (1994). Expert network: effective and efficient learning from human decisions in text categorization and retrieval. In: *SIGIR '94 Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Springer, New York, pp. 13–22.

Yang, Y. (1999). An evaluation of statistical approaches to text categorization. *Inf. Retr.*, 1(1-2), 69–90.

Yang, Y.. Chute, C. (1992). A linear least squares fit mapping method for information retrieval from natural language texts. In: *Proceedings of the 14th Conference on Computational Linguistics*, pp. 447–453.

Yang, Y., Pedersen, J. (1997). A comparative study on feature selection in text categorization. In: *ICML '97 Proceedings of the Fourteenth International Conference on Machine Learning*. Kaufmann, Los Alfos, pp. 412–420.

V. Balys graduated from Vilnius University in 2004 receiving master of science of informatics degree. He is a doctoral student in the Vilnius University Institute of Mathematics and Informatics. His PhD research topic is statistical classification methods and their applications in the classification of scientific texts.

R. Rudzkis graduated from the Kaunas Polytechnical Institute in 1973. He received the PhD degree from Vilnius University (1978), the dr. habil. from the Vilnius University Institute of Mathematics and Informatics (1993). He was conferred the title of professor at Vytatutas Magnus University in 1996. Currently, he is a senior researcher in the Department of Probability Theory and Statistics in the Vilnius University Institute of Mathematics and Informatics. His research interests include probability theory and mathematical statistics models with applications, econometric modelling.

Statistinis mokslo publikacijų klasifikavimas

Vaidas BALYS, Rimantas RUDZKIS

Straipsnyje nagrinėjamas automatinio mokslo tekstų klasifikavimo uždavinys. Tiriami statistine mokslo terminijos pasiskirstymo tekstuose analize paremti metodai. Pateikiamos informatyviausių terminų nustatymo ir papildomos kontekstinės informacijos, susijusios su terminų pozicijomis tekste, panaudojimo klasifikavime procedūros. Pristatomi nagrinėtų ir alternatyvių algoritmų eksperimentinio tyrimo, atlikto realių publikacijų duomenų bazių pagrindu, rezultatai.