Feature Set Reduction for Document Classification Problems

Karel Fuka, Rudolf Hanka
Medical Informatics Unit
University of Cambridge
Robinson Way
Cambridge
CB2 2SR
UK
{kf218,rh10}@cam.ac.uk

Abstract

With a growing amount of electronic documents available, there is a need to classify documents automatically. In growing text classification applications, important-term selection is a critical task for the classifier performance. Although many different techniques and heuristics have been developed, this paper shows that many of them are just a sub-set of more advanced methods originating in the field of pattern recognition. The paper puts these techniques into the pattern recognition context. It also shows that despite of some theoretical problems in this area, which are identified and described, pattern recognition techniques might be found useful for text classification tasks. The performance of different feature set reduction techniques is measured by classification accuracy when different numbers of features are selected/extracted. Results for different numbers of features and various techniques are then compared and analysed.

1. Introduction

Unstructured information sources have drawn recently more attention mostly because of a rising number of electronic documents accessible through different sources like e-mails, huge digital libraries, local networks, but most significantly via WWW. Researchers from many different fields try to use their own techniques to automatically organise these data collections and enable users to access data in some informed way, i.e. users know how to navigate through these data sources and understand the organisational structure without a priori organising those data. One of the techniques usually employed is a classification, which enables automatic routing of a particular document into some pre-specified sub-collection (see for example [Rennie, 2000]).

This paper proposes some challenging research problems that can be found in the area of text classification and then concentrates on the feature set reduction methodology as one of the key topics. Different existing methods for feature set reduction have been developed in the areas of information retrieval and further in text classification. Although these techniques have been independently developed over many years, they have a strong relationship with methods from pattern recognition area where the methodology seems to have reached more complex theoretical results. The paper therefore aims to put special text-oriented techniques into the context and terminology developed in pattern recognition.

Experimental results compare different feature set reduction methods and illustrate how the use of some well-known pattern recognition methods can improve classification accuracy.

2. Document Classification

To be able to classify documents, one must find a way how to reasonably simply represent documents in a way that this representation preserves as much of the original information as possible and also is simple enough from a computational point of view. Different ways of representing documents that reflect different needs of their users have been proposed.

The simplest method called *bag of words* used in the vast majority of current applications is based on the application of basic terms (either all of them or a subset like

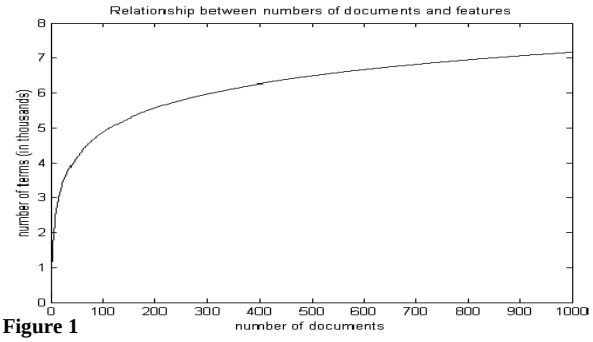

Figure 1

nouns). It is also used in this paper. Many other representations have been found, which behave better for some special purposes. For example *conceptual features* (represent meaning of the original documents), c*ontextual features* (contain contextual information of terms e.g. bigrams, trigrams, or more sophisticated noun-bigrams [Jensen and Martinez, 2000]), *mechanically extracted features* (extracted from documents without using any knowledge about its content or language structure, possibly based even on a compressed version of the original document [Fontaine and Matwin, 2000]), *document structure features* (a total number of words, number of sentences, average length of sentences, etc. used in the area of computer forensics by de Vel [2000]).

The most general and widely used *bag of words model* can be described as follows. Let $t_1,t_2,...,t_n$ denote distinct terms used for indexing documents and $D_1,D_2,...,D_m$ documents. Document D_i is represented by a term vector defined as:

$$\boldsymbol{D}_i = (a_{i1},a_{i2},...,a_{in})^T \qquad (1)$$

where a_{ij} is a weight of a term t_j in the document D_i. The values a_{ij} can be just simple frequencies of the term t_j in the document D_i or they are further normalised. This representation will also be used in the further text.

For classification purposes, the previous model is extended and every document is represented as a tuple:

$$D'_i = <\boldsymbol{D}_i, c_k> \qquad (2)$$

where $\boldsymbol{D}_i$ is a previously defined document vector and c_k, for $k=1, \ldots, K$, is its class. As defined now, this problem becomes a standard classification task and one could possibly approach it by the application of many existing pattern recognition methods. But, as more detailed analysis shows, there are some special aspects in this area that need more attention.

3. Curse of Dimensionality

Curse of dimensionality is a phenomenon, whose original meaning was published in [Bellman, 1961] and refers to the fact that some problems are very difficult to compute because of a number of features and the solutions often exceed available computing resources. This phenomenon in information retrieval domain has been first mentioned in [Koller and Sahami, 1997].

Gradually, the curse of dimensionality has also begun to denote another phenomenon closely related to high-dimensional data. In the highly dimensional vector spaces, data are extremely sparse and so to estimate any parameter, one must have many samples to achieve a reasonable accuracy level. The dimensionality of vector space increases approximately more than exponentially with the number of samples. Moreover, this again severely restricts possible applications because the resulting computer power demand is too high and heavily restricts a potential set of solutions. Of course, this fully applies to the area of document classification, as is mentioned in [Zervas, 1999].

Moreover, in the real-world document classification applications, it is hardly possible to collect a sufficient set of documents. This is illustrated in Figure 1 that is based on the data used in the experiments described later in detail. The graph shows that with the increasing number of data vectors (documents), number of features (terms) also increases. This phenomenon is caused by the methodology used for building document vectors – feature set is built up over the terms that are used in a particular set of documents. Therefore, it is highly likely that by adding new documents into a set, one gets additional terms and so the number of features increases. One hypothetic solution would be to take all the possible terms in a language and use these as features from the beginning. Assuming one would be able to do that (at least for a pre-specified subject domain, for example medical domain), there would be at least two new problems one would have to overcome:

1. Assuming approximately one hundred thousand terms (features) in order to get reasonable estimates, one would need to have more then say at least the same number of documents, but usually twice. This is not always possible.
2. If any of the terms does not occur in any documents, its feature value is always zero and the variance is zero as well. This feature would not add any information for classification, not mentioning possible singularities caused by this in some statistical classifiers.

Thus it is obvious that in the area of document classification it is highly likely that one will have to face a situation when there are more features than observations. First of all, it must be stressed that there have been some studies [Duin, 2000] that showed that some classifiers may have a low generalisation error even in cases when the number of data vectors is lower than the number of features. This is called peaking phenomenon (for more details see [Raudys and Duin, 1998]).

Furthermore, there are techniques for dimensionality reduction that may lead to an improved generalisation and more stable results (in sense of variance). There are even many of them particularly created for textual documents. These techniques are described in the following chapter.

4. Feature Set Reduction Techniques

In pattern recognition area, methods for dimensionality reduction are divided into two categories [Fukunaga, 1972]:

- *Feature selection* – the dimensionality is reduced by selecting a subset of original features. The removed features are not used in the computation anymore. The aim of feature selection methods is to determine a subset of *d* features from the set of *m*, for which a criterion *J* will be maximised.
- *Feature extraction* – the original vector space is transformed into a new one with some special characteristics and the reduction is made in a new vector space. Comparing to feature selection, all data features are used. In this case, they are just transformed (using a linear or non-linear transformation) to a reduced dimension space with the aim of replacing the original features by a smaller set of underlying features.

Both of these approaches require optimisation of some criterion function *J*, which is usually a measure of distance or dissimilarity between distributions [Pudil *et al.*, 1994a]. In document classification area, special techniques have been developed, which are based on the domain knowledge and defining criterion *J*, which is not usually based on distance measurements.

Following methods for term-number reduction in text area could be called as *feature selection* methods:

- *Threshold methods* - threshold methods are based on removal of features, whose frequencies are greater than (or less than) a defined threshold value. These methods are currently most popular because they are reasonably fast and efficient. On the other hand, they completely ignore the existence of other features and evaluate every feature on its own. This may lead to a problem, because individually the best terms may give worse classification results than another group of features, which need not be the best features individually. This is called *feature nesting* (see [Pudil *et al.*, 1994b]). Typical examples of threshold methods are: *document frequency thresholding*, *information gain*, *mutual information*, χ^2 *statistic*, *term strength*, *odds ratio*, *weirdness coefficient*. For more detailed description see e.g. [Yang and Pederson, 1997], [Galavotti *et al.*, 2000], [Mladenic, 1998], [Ahmad, 1994].
- *Information theory methods* - In information theory, the least predictable terms carry the greatest information value [Gudivala *et al.*, 1997]. The least predictable terms are those that occur with the smallest probabilities. Information theory concepts have been used to derive a measure, called *signal-noise ratio*, of term usefulness for indexing. This method favours terms that are concentrated in particular documents. Therefore its properties are similar to threshold methods described in the previous paragraph.

From *feature extraction* methods, at least one that has found some use in the document classification is:

- *Feature clustering* - the aim is to find groups of similar features (or in other words, features that have the same or similar function in the vector space) and group them together. A group (or cluster) is forming a new feature, which is also sometimes called *concept*. The method has been originally applied to a thesaurus induction, where the idea is to build a thesaurus automatically from a corpus [Roussinov and Chen, 1998]. Such a thesaurus contains terms grouped into clusters. The resulting clusters can be viewed as a kind of concepts connecting similar terms together and such concepts can be used for forming document vectors.

It is important to stress that the above methods have been developed independently of any pattern recognition methods and are usually more heuristic than those used in pattern recognition, and often originate from information retrieval area. The experimental section of this paper shows how these methods compare with those from the "classical" pattern recognition area.

5. Comparison of Different Methods

To find out how different feature space reduction techniques affect the classification of documents, there have been conducted some experiments that illustrate this behaviour. The experiments were based on N=408 documents from k=9 different classes of Reuters-21578 collection. Two thirds of documents were randomly selected to comprise a training set and remaining one third was used for testing. After the removal of stop words and application of stemming, there remained D=3822 terms (or more precisely stems). Original frequency values were normalised:

$$a_{ij} = t_{ij} \log(N / m_j) \qquad (3)$$

where t_{ij} - number of occurrences of term t_j in i-th document (term frequency), m_j - number of documents indexed by term t_j, N - number of all documents.

Different algorithms were used to select or extract a particular subset of d features from the set of all D features and classification accuracy was recorded for every different method on the same level of feature numbers. The classifier for *d = 6, 12, 25, 50, 100, 200, 400, 600, 800 out of D = 3822* remained the same for all the experiments to minimise any impact on presented results. Classifier and different feature set reduction methods are described in the next paragraphs followed by the detailed description of experiments.

5. 1. Methods

χ^2 Statistic

The χ^2 statistic is originally used in the statistical analysis of independent events. Its application in feature selection is straightforward. One constructs a contingency table for term and category and the χ^2 statistic is calculated as:

$$\chi^2(t, c_i) = N(P(t, c_i)P(t', c_i') - P(t',c_i)P(t,c_i'))^2 \\ (\ P(t, .)\ P(t', .)\ P(. , c_i)\ P(\ . ,c_i')\ )^{-1} \quad \text{for } i = 1,\ldots, k, \quad (4)$$

where N is the total number of documents, P(., x) and P(x, .) denote marginal probabilities. t' means that term t is not present and c_i' denotes all categories, but c_i.

To find out which feature should be taken, following characteristic was calculated for every term:

$$\chi^2_{avg}(t) = \sum_i P(c_i)\,\chi^2(t, c_i), \text{for } i = 1,\ldots, m \quad (5)$$

where $P(c_i)$ is a prior probability of class c_i. Then the values were sorted in the ascending order and a set of d top terms was built up.

Clustering

Before the detailed description of various clustering methods, some principles of term clustering need to be explained. As stated in the previous text, clustering is a kind of feature extraction. The basic idea is to find out new variables $\xi_1, \xi_2, \ldots, \xi_d$ so that:

$$\xi_k = \Sigma_i \Sigma_j\, a_{ij},\ i=1, \ldots, N;\ j \in S_l,\ l=1, \ldots, d \quad (6)$$

where a_{ij} is a frequency of j-th term in i-th document and S_l is a set of positive whole numbers denoting indexes of terms that are grouped together and participate by their frequencies to ξ_k. Sets S_l are created automatically by a clustering algorithm as follows. Let A be a matrix in which rows are particular observations (documents) and columns are features (terms). Let A^t be a transposition of this matrix to which we apply using a clustering algorithm. Then resulting clusters are called *concepts*. They can be seen as synonyms of terms from the information retrieval point of view because they can be found in similar documents with similar frequency distribution and therefore they would lead to the retrieval of the same group of documents if they had been used individually as query terms. These principles are common to all clustering algorithms employed on the term clustering task. A description of Kohonen's self organising map (SOM) applied as one of the clustering algorithms followed be explanation of hierarchical clustering methods.

SOM Clustering. Kohonen's Self-Organising Maps (SOMs) (as described in [Honkela, 1997]) form a very important approach in the area of clustering techniques. The Self-Organising Map can be visualised as a neural network array and its functionality is usually equaled to processes found in the human brain's cortex neurons. The nodes of this array become specifically tuned to various input signal patterns in an orderly fashion. In the case of this study, the SOM network had hexagonal architecture. The number of neurons corresponds to the number of clusters – concepts – that are due to be found. Because the network architecture is two dimensional, configurations for different numbers of clusters differed: 800 was represented as 200 by 40 neuron configuration, 600 was 30x20, 400 as 20x20, 200 as 20x10, 100 as 10x10, 50 as 25x2, 25 as 5x5, 12 as 4x3, and 6 as 3x2.

Hierarchical Clustering. Hierarchical tree clustering is based on the idea of building a hierarchy of objects based on the similarity between groups of objects, see for example [Chung and Chen, 1994]. In this case, similarity between terms or term clusters is based on the Euclidean distance between clusters x and y:

$$distance(x,y) = \{\Sigma (x_i - y_i)^2\ \}^{1/2} \quad (7)$$

To calculate a distance between two clusters, the *single-link* clustering was applied. It attempts at each step to join nearest pair of objects or clusters to join them and make up a new cluster. Therefore, clusters are nested in the form of a tree. To get d concepts, one must get to a level of the tree, on which d clusters can be found. These clusters form new concepts used as features in the classification process.

Hierarchical Clustering after Removal of the Most Frequent Cluster. Empirical results show that generated clusters are not equivalent to each other when the number of terms associated with every cluster is concerned. On level l there exist d clusters where one of those clusters contains approximately $n-d+1$ terms. This phenomenon can be interpreted as follows. The substantial majority of terms are 'similar' in the sense of Euclidean distance. These terms tend to be put into a cluster very early. Approximately $d-1$ remaining terms are very different and therefore it can be assumed that they contain a lot of significant information for document classification. Under this assumption, we can freely remove the cluster containing $n-d+1$ terms and use only the remaining clusters as newly formed features.

Principal Components Analysis (PCA)

The purpose of principle components analysis is to derive new variables that are linear combinations of the original variables and are uncorrelated. Geometrically, principal components analysis can be thought of as a rotation of the axes of the original coordinate system to a new set of orthogonal axes that are ordered in terms of amount of variation of the original data they account for [Webb, 1999].

Let $\boldsymbol{x}$ be the original D dimensional observation vector representing a document and $\boldsymbol{\xi}$ is a new d-dimensional extracted vector obtained by a linear transformation:

$$\boldsymbol{\xi} = A'\,\boldsymbol{x} \quad (8)$$

The coefficients of D by d matrix A can be found as follows. Let Σ be the covariance matrix of $\boldsymbol{x}$. It has D eigenvalues:

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_D \geq 0 \quad (9)$$

It can be deduced that matrix A is constituted of d eigenvectors corresponding to d largest eigenvalues λ_i, $D \geq i \geq 1$. Every eigenvector is a column of the matrix A.

5. 2. Classifier Used

To compare different feature set reduction methods, one must stick with a classifier to make results comparable. We selected a multilayer feedforward neural network as a flexible and well-established classifier.

Multilayer feedforward neural network had 100 neurons in hidden layer and 9 neurons in the output layer. The tangent sigmoid function was used as a transfer function in every neuron. Gradient descent was employed as a training function. Every output neuron was associated with one class – therefore when input vector (representing a document) was from *n*-th class, the training value for the *n*-th output neuron was 1 and the remaining values were –1.

The actual number of input nodes depended on the testing because for different numbers of features *d* training and classification had to be repeated.

5. 3. Experiments

All experiments measured the impact of feature reduction methods on classification accuracy defined as:

$$accuracy = (1 - \mu / N_t) * 100\% \qquad (\mathbf{10})$$

where μ is a number of misclassified documents from a testing set containing N_t documents. Every result represents a single run of the classifier.

Experiment A

In this experiment different feature reduction methods were applied on the set of original 3822 terms and their impact on the classification accuracy was measured for sets of different sizes. Figure 2 contains main results.

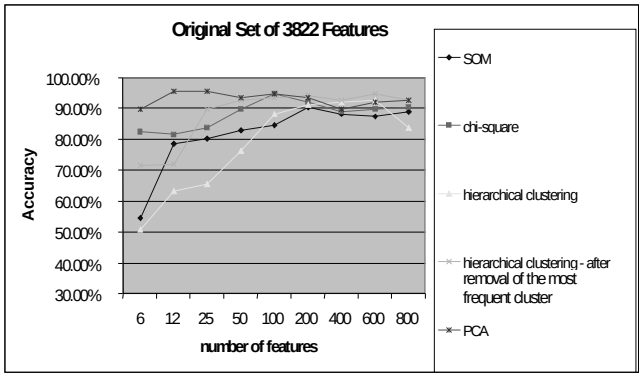

Figure 2

The only feature selection method employed in the test, χ^2 statistic, proved to give stable and relatively good results of accuracy between approx. 81% and 95%. On the other hand, the performance of different clustering methods varies significantly. Hierarchical clustering brought very good results when the actual number of created clusters was larger than 100. At this point, this method was comparable with all others. When the most frequent cluster was removed, then the hierarchical clustering gave reasonable results, but for fewer than 25 clusters accuracy fell to approx. 70%. Results of SOM were very similar to those reached by the hierarchical clustering, but relatively more stable. Furthermore, as far as stability is concerned, results were very similar to χ^2, but never exceeded 90% accuracy level. Finally, PCA seems to be a very good method for feature extraction. It gave very good accuracy even for low numbers of extracted features. Also, using this method, the best ever results of accuracy – more than 95% – have been obtained.

Experiment B

In this experiment, the focus was put on pre-processing phase and feature reduction methods were compared again. First, the original data set was reduced so that only nouns detected by a statistical parser remained in documents. Then the stemming was carried out and the resulting feature set

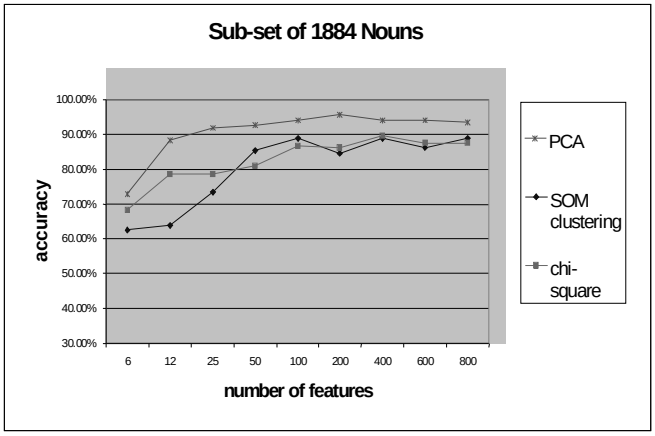

Figure 3

contained *D*=1884 features.

Results are plotted in Figure 3. In this case, three methods have been compared. Generally speaking, one has to use more features regardless of the method used for the feature space reduction because all methods get much worse results when fewer than 25 features were employed. This can be explained the loss of information when all grammar categories, except nouns are left out of the feature set. It also means that other grammar categories have a descriptive value and should not be completely ignored.

As far as particular feature set reduction methods are concerned, χ^2 and SOM performed in a very similar fashion in this case. On the other hand, PCA outperformed both methods and is the only method that reached classification accuracy results higher than 90%.

Experiment C

In this experiment, the emphases were put onto the PCA as the best performing method. The aim was to find out how PCA behaves when different selection methods are employed. 300 features have been selected from the same set defined in experiment 2. The way applied for selection of those 300 features out of 1884 obviously affects the results as is shown in detail. Firstly the 300 hundred features had been selected via χ^2 method. PCA was then performed to form sub-sets of 200, 100, 50, 25, 12, and 6 features used for subsequent classification. Then 300 features were randomly selected and PCA was again performed in the same manner Finally, the sub-sets of the same sizes were selected randomly and PCA from the Experiment B was added for comparison. This is illustrated in Figure 4.

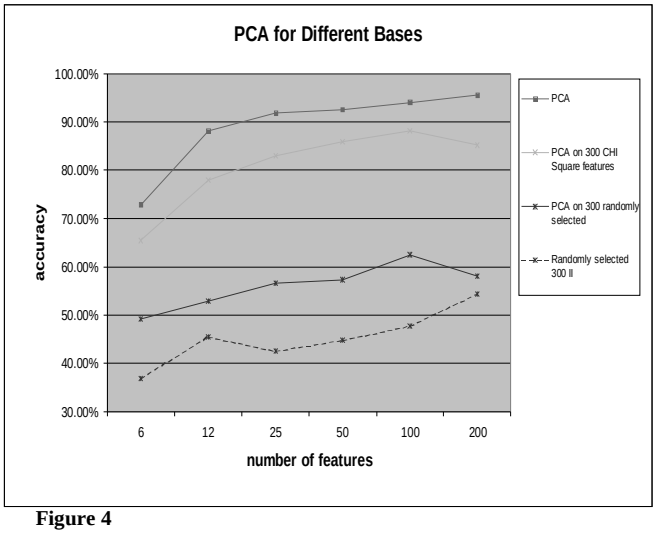

Figure 4

It is obvious that the initial random sub-set selection removed important information because PCA performed on the original set (data series denoted as PCA) gave significantly better results than PCA performed on the 300 randomly selected features. When 300 features have been selected via χ^2, PCA performed on this subset gave the second best results after PCA performed on the whole set.

Not surprisingly, all feature reduction methods outperformed blind random selection of features. This fact is illustrated by the performance of randomly selected subsets whose classification accuracy hardly reaches 50% level.

6. Conclusions

Many different techniques for removing 'less descriptive' terms have developed in the area of information retrieval and text mining. These methods usually use some knowledge about the domain as well as some heuristics to obtain relatively good results. We have shown that all these techniques can be systematically classified as feature set reduction methods and a clear relationship between these techniques and techniques from the pattern recognition area can be found.

It also turns out that applying this new knowledge and using some "classical" pattern recognition methods in the text domain, the classification results can be improved. This was demonstrated when we used principal component analysis for feature extraction. It performed quite well on all the tests.

On the other hand, χ^2 statistics still gave very good results and should not be ignored.

Clustering methods seem to achieve results with a high variability. For numbers of features greater than 100, they can lead to the classification accuracy levels similar to those of χ^2 or even PCA. But under this level, the classification performance significantly drops.

As far as different preprocessing techniques are concerned, the usage of nouns only did not affect the overall performance in a significant manner. Only at very low numbers of features the performance of PCA and χ^2 dropped under 80%. But for higher numbers features, this did not have any significant impact. This implies that (at least for this data set) nouns can bear a reasonable amount of information that is used for classification.

Finally, all the feature set reduction methods perform much better than a blind random selection of features. Though one would assume this fact at the beginning, it just stresses how important it is to select a proper method for the feature set reduction.

Acknowledgments

Data used for the analysis have been selected from the "Reuters-21578, Distribution 1.0" document collection. Mr. Karel Fuka is financially supported by Cambridge Overseas Trust, U.K.

References

[Ahmad, 1994] Ahmad, K.: Language engineering and the processing of specialist terminology. European Network in Language and Speech, Edinburgh, 1994, pp. 9-16

[Bellman, 1961] Bellman, R.: Adaptive Control Processes: A Guided Tour. Princeton University Press, 1961

[Chung and Chen, 1994] Chung-hsin – Chen, H.: An automatic indexing and neural network approach to concept retrieval and classification of multilingual (Chinese-English) documents. IEEE Transactions on Systems, Man, and Cybernetics, 1994, 29 pages

[Duin, 2000] Duin, R.P.W.: Classifiers in Almost Empty Spaces. In Pattern Recognition and Neural Networks, IEEE Computer Society Press, vol. 2, Los Alamitos, 2000, pp. 1-7

[Fontaine and Matwin, 2000] Fontaine, M. – Matwin, S.: Features extraction techniques of unintelligible texts. KDD-2000 Workshop on Text Mining, Boston, 2000, pp. 95-96

[Fukunaga, 1972] Fukunaga, K.: Introduction to Statistical Pattern Recognition. Academic Press, New York, 1972

[Galavotti *et al.*, 2000] Galavotti, L. – Sebastiani, F. – Simi, M.: Feature selection and negative evidence in automated text categorization. Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries, ECDL-00, Lisbon, 2000

[Gudivala *et al.*, 1997] Gudivada, V. N. – Raghavan, V. V. – Grosky, W. I. – Kasanagottu, R.: Information retrieval on the World Wide Web. In IEEE Internet Computing, Vol. 1, No. 5, September, 1997, pp. 58-68

[Honkela, 1997] Honkela, T.: Self-Organizing Maps in Natural Language Processing. Ph.D. Dissertation, University of Technology, Helsinki, 1997

[Jensen and Martinez, 2000] Jensen, L. S. – Martinez, T.: Improving text classification by using conceptual and

contextual features. KDD-2000 Workshop on Text Mining, Boston, 2000, pp. 101-102

[Koller and Sahami, 1997] Koller, D. – Sahami, M.: Hierarchically classifying documents using very few words. In Proceedings of International Conference on Machine Learning, 1997, pp. 170-178

[Mladenic, 1998] Mladenic, D.: Feature selection in text-learning. Proceedings of 10th European Conference on Machine Learning, 1998

[Pudil *et al.*, 1994a] Pudil, P. - Novovičová, J. - Ferri, F.: Methods of dimensionality reduction in statistical pattern recognition. In Proceedings of the IEE European Workshop CMP '94, Prague, Institute of Information Theory and Automation, 1994, pp. 185-189

[Pudil *et al.*, 1994b] Pudil, P. - Novovičová, J. - Kittler, J.: Floating search methods in feature selection. In Pattern Recognition Letters, 15, 1994, pp. 1119-1125

[Raudys and Duin, 1998] Raudys, S. – Duin, R.P.W.: On expected classification error of the Fisher linear classifier with pseudo-inverse covariance matrix. In Pattern Recognition Letters, vol. 19, no. 5-6, 1998, pp. 385-392

[Rennie, 2000] Rennie, J.: ifile: An application of machine learning to e-mail filtering. KDD-2000 Workshop on Text Mining, Boston, 2000

[Roussinov and Chen, 1998] Roussinov, D. – Chen, H.: A scalable Self-organizing map algorithm for textual classification: A neural network approach to thesaurus generation. In Communication and Cognition – Artificial Intelligence, Vol. 15, No. 1-2, 1998, pp. 81-112

[Vel, 2000] de Vel, O.: Mining e-mail authorship. KDD-2000 Workshop on Text Mining, Boston, 2000, pp. 21-27

[Webb, 1999] Webb, A.: Statistical Pattern Recognition. Arnold publishing, London, 1999

[Yang and Pederson, 1997] Yang, Y. – Pedersen, J. O.: A comparative study on feature selection in text categorisation. In Proceedings of the 14th International Conference on Machine Learning, ICML-97, 1997, pp. 412-420

[Zervas, 1999] Zervas, G. – Rüger, S.M.: The curse of dimensionality and document clustering. In Proceedings of the IEEE Searching for Information: AI and IR Approaches, 1999