INFORMATION
SYSTEMS IN
MANAGEMENT

Information Systems in Management (2016) Vol. 5 (1) 24–35

EFFECTIVE MULTI-LABEL CLASSIFICATION METHOD WITH APPLICATIONS TO TEXT DOCUMENT CATEGORIZATION

KINGA GLINKA, DANUTA ZAKRZEWSKA

Institute of Information Technology, Lodz University of Technology

Increasing number of repositories of online documents resulted in growing demand for automatic categorization algorithms. However, in many cases the texts should be assigned to more than one class. In the paper, new multi-label classification algorithm for short documents is considered. The presented problem transformation Labels Chain (LC) algorithm is based on relationship between labels, and consecutively uses result labels as new attributes in the following classification process. The method is validated by experiments conducted on several real text datasets of restaurant reviews, with different number of instances, taking into account such classifiers as kNN, Naive Bayes, SVM and C4.5. The obtained results showed the good performance of the LC method, comparing to the problem transformation methods like Binary Relevance and Label Powerset.

Keywords: Multi-label Classification, Text Categorization, Problem Transformation Methods, Text Management

1. Introduction

Text document categorization is an important task, playing significant role in such areas as information retrieval, text management, web searching or sentiment analysis. However, in many cases documents should be assigned to more than one class. Then, multi-label classification, which contrarily to the single-label one aims at predicting more than one predefined class label, can be used.

Multi-label classification for text documents have to deal with multidimensional datasets of many attributes. In many cases document datasets contain relatively small number of instances, at the same time. Such situation can take place in the case of medical records or documents from narrow specialized domains. Text documents are usually described by many attributes what makes the process of multi-label classification more complex and thus, methods dealing with that kind of data seem to be necessary. There exist several techniques for multi-label classification that can be used for any dataset. However, they do not provide satisfactory accuracy in many cases, especially when sets of attributes are large.

In the paper, application of the problem transformation method, which deals with multi-label classification when the number of attributes significantly exceeds the number of instances, is considered. The method was firstly introduced in [1], where its performance was examined by taking into account accuracy for 2 label classification of datasets of images and music. In the current paper, we propose to use the technique for text document datasets, taking into account more labels. The technique is validated by the experiments conducted on datasets of different number of instances and attributes, taking into account not only classification accuracy but also *Hamming Loss* measure [2]. The results are compared with the ones obtained by application of the most commonly used methods: *Binary Relevance* [3, 4, 5] and *Label Power-set* [4, 5, 6, 7].

The reminder of the paper is organized as follows. In the next section relevant research concerning multi-label classification of text documents dataset is presented. Then the proposed approach together with evaluation measures are described. In the following section the experiments and their results are depicted. Finally some concluding remarks and future research are presented.

2. Relevant research

Many techniques of multi-label classification have been proposed so far. However, there are two main approaches, which are the most commonly applied. The first one is based on adaptation methods, which extend specific algorithms to obtain the classification results directly. The second approach is independent of the learning algorithms and transforms multi-label classification problem into single-label tasks. Then well-known classification algorithms can be applied.

There exist several transformation techniques [4]. As the most popular ones there should be mentioned *Binary Relevance* and *Label Power-set* techniques. The first method converts multi-label problem into several binary classification problems by using one-against-all strategy. Its main disadvantage consists in ignoring label correlations which may exist in a dataset (see [3, 4, 5]). *Label Power-set* method creates new classes of all unique sets of labels which exist in the multi-label training data. Thus, every complex multi-label task can be changed into one single-label classification. Therefore, this method can be used regardless of number and variety of labels assigned to the instances. The main disadvantage of creating new labels is that it may lead to datasets with a large number of classes and only few instances representing them [4, 5, 6, 7].

Text categorization is one of the main domain, where multi-label classification is applied, however most of the researchers examined the proposed approaches taking into account datasets of different characters [4, 5]. Multi-label text classification was considered by Shapire and Singer [2], who introduced the boosting method, which consists in combining inaccurate rules into the single accurate one. They considered the cases of text documents with small number of categories. Their approach was further developed in the papers [8, 9].

Fuzzy approach was proposed by Lee and Jiang [10]. They used a fuzzy relevance measure to reduce the number of dimensions and applied clustering to build region of categories.

3. Materials and methods

3.1. Proposed approach

The proposed transformation methodology is based on separate single-label classification tasks. Two methods are considered: *Independent Labels* with all the tasks applied individually and *Labels Chain* which takes into account consequential labels in each succeeding classification process. Let L be the set of all the labels and let K denote a set of labels relevant for an instance.

Independent Labels (IL) is the approach, where each label constitutes a separate single-label task. *IL* algorithm works similarly to *Binary Relevance* method, however, it requires to learn $|K|$ multiclass classifiers, instead of $|L|$ binary classifiers. Such approach makes the method competitive in time and computational complexity in the cases of the small number of labels per instance. The main assumption concerns known number of labels for instances. Unfortunately, the algorithm ignores existing label correlations during classification process, what may result in losing some vital information and may provide poor prediction quality in some cases.

Labels Chain (LC) is the improvement of *IL* method, that uses mapping of relationship between labels. New proposed algorithm also requires to learn $|K|$ multiclass classifiers, but this one, in contrast to *IL,* consecutively uses result labels as new attributes in the following classification process. It creates the classifications chain (the idea has been used so far only for binary classifications [11]), taking into account that classifications are not totally independent from themselves, what enables providing better predictive accuracy. This feature is especially important in multi-label problems with small number of labels K, because in these cases the value of a new, added attribute is more significant for classification process. The *Labels Chain* method can be also applied taking into account different order of classifications, with $|K|!$ available order combinations. As in *IL*, the number of labels for instances is assumed to be known.

In further considerations, *Independent Labels* is used as indirect method, improved by *Labels Chain* approach. Comparison of results from both algorithms during experiments shows the advantage of using relationship between labels. The obtained results are also compared with those got by the most popular *Binary Relevance* and *Label Power-set* algorithms, taking into account two evaluation metrics.

3.2. Evaluation metrics

Hamming Loss was proposed in [2] for evaluating the performance of multi-label classification, it calculates the fraction of incorrectly classified single labels to the total number of labels. Since it is a loss function, its smaller value is connected with the better performance of the algorithm. It is defined as:

$$Hamming\ Loss = \frac{1}{N}\sum_{i=1}^{N}\frac{xor\left(Y_i, F\left(x_i\right)\right)}{\left|L\right|} \tag{1}$$

where: x_i are instances, $i = 1\ldots N$, N is their total number in the test set, Y_i denotes the set of true labels and $F(x_i)$ is a set of labels predicted during classification process, and operation $xor(Y_i, F(x_i))$ gives difference between these two sets.

Classification Accuracy (also known as *exact match*) is much more strict evaluation metric for multi-label classification. Contrarily to the *Hamming Loss* measure, it ignores partially correct sets of labels by marking them as incorrect predictions, and requires all labels to be an exact match of the true set of labels. *Classification Accuracy* for multi-label classification is defined as [12]:

$$Classification\ Accuracy = \frac{1}{N}\sum_{i=1}^{N} I\left(Y_i = F\left(x_i\right)\right) \tag{2}$$

where $I(true) = 1$ and $I(false) = 0$.

3.3. Text document datasets from Yelp

In order to evaluate the proposed method, several real text datasets of restaurant reviews from Yelp website [13] – the online business directory were considered. Yelp users give ratings and write reviews about local businesses and services on Yelp. These reviews are usually short texts with about hundred words, which are to help other users to make choice of restaurants, shopping mall, home service and others. In many cases, the reviews describe various aspects and experiences connected with the considered business.

Restaurant reviews from Yelp can be classified into five predefined categories: *Food*, *Service*, *Ambience*, *Deals/Discounts* and *Worthiness*. Interpretation of

Food and S*ervice* categories seems to be obvious. *Ambience* refers to the look and feel of the place. *Deals* and *Discounts* correspond to offers during happy hours, or specials run by the restaurant. Finally, *Worthiness* can be interpreted as value for money and is different from the price attribute already provided by Yelp. All the categories were introduced and analyzed in [14]. As each review can be associated with multiple categories at the same time, its categorization can be considered as multi-label classification problem. Such approach seems to be very effective in making a decision, because it helps in understanding why the reviewer rated the restaurant *low* or *high*. Moreover, it avoids wasting time reading reviews that do not relate to the category, which user is interested in. Although, the described functionality is useful for any kind of business, the scope of our investigations will be limited only to restaurants.

The basic data corpus comes from [14] and contains instances described by 668 attributes – 375 unigrams, 208 bigrams and 120 trigrams [15]. Such approach based on keywords allows to present text of a review as a vector of features.

There were taken into account datasets with different number of instances and different number of assigned labels for instances. There were considered 6 main datasets randomly selected from all the data:

- 3 datasets of 1676 instances, with two labels assigned (named *TwoLabels_1*, T*woLabels_2* and *TwoLabels_3*),
- 3 datasets of 1200 instances, with three labels assigned (named similarly *ThreeLabels_1*, *ThreeLabels_2* and *ThreeLabels_3*).

The datasets were used to create the ones of the smaller number of instances. From each of the dataset half of the instances were randomly selected. This process was consecutively repeated several times for newly created datasets. Thus, from the datasets of 1676 instances we obtained new ones of 838, 419, 210, 105 and 53 records, and the ones of 1200 instances were respectively reduced to 600, 300, 150, 75 and 38 objects. That way, there were obtained 36 datasets of different number of instances, part of them with the number of attributes, which significantly exceeds relatively small number of instances.

4. Experiment results and discussion

The aim of the experiments was to examine the performance of the proposed technique comparing to the commonly used problem transformation methods. The experiments were carried out on all the datasets described in the Section 3.3. Values of *Classification Accuracy* and *Hamming Loss* measures were compared for considered methods: *Binary Relevance (BR), Label Power-set (LP)*, and investigated *Independent Labels (IL)* and *Labels Chain (LC)*. In the case of the *LC* technique, different possible label orders were examined. The final results were indicated according to the best accuracy values. The experiments were conducted for the well-known one-label classifiers: *k*-nearest neighbors, naive Bayes, support vector machine SVM and C4.5 decision tree [16], which were conjunct with the considered problem transformation methods.

The software implemented for experiments was based on WEKA Open Source [17] with default parameters of WEKA software, and was running under Java JDK 1.8, on 64-bit machine with a dual core processor. In a classification process, each of a single dataset was divided into two parts – training set (60% of instances) and test set (40% of instances).

Values of *Hamming Loss* measure for all the tested datasets with assigned 2 labels *TwoLabels_1*, T*woLabels_2* and *TwoLabels_3* are presented in Tab. 1. Tab. 2, Fig. 1 and Fig. 2 show Classification Accuracy values for all the datasets. In all the tables the best results in rows are shadowed, taking into account all the considered methods: *Binary Relevance* (*BR*), *Label Power-set* (*LP*), *Independent Labels* (*IL*) and *Labels Chain* (*LC*), for different classifiers and dataset sizes. Considered classifiers are marked in the tables with the following abbreviations: *k*-nearest neighbors *kNN*, naive Bayes *NB*, support vector machine *SVM* and *C4.5* decision tree.

Table 1. Datasets with 2 labels assigned – results of Hamming Loss

Instances, classifiers		**TwoLabels_1**				**TwoLabels_2**				**TwoLabels_3**			
		BR	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**
1676	kNN	0.230	0.233	0.232	0.253	0.288	0.276	0.289	0.305	0.251	0.256	0.256	0.244
	NB	0.228	0.229	0.233	0.252	0.236	0.232	0.240	0.237	0.227	0.208	0.223	0.222
	SVM	0.281	0.281	0.281	0.276	0.281	0.281	0.279	0.273	0.279	0.279	0.279	0.278
	C4.5	0.245	0.265	0.257	0.277	0.219	0.259	0.272	0.270	0.239	0.269	0.270	0.271
838	kNN	0.312	0.314	0.314	0.281	0.282	0.284	0.282	0.258	0.302	0.300	0.309	0.302
	NB	0.261	0.267	0.281	0.278	0.234	0.227	0.241	0.243	0.235	0.235	0.242	0.248
	SVM	0.273	0.273	0.273	0.259	0.281	0.281	0.281	0.281	0.302	0.302	0.302	0.281
	C4.5	0.269	0.287	0.297	0.301	0.244	0.294	0.272	0.281	0.257	0.294	0.286	0.324
419	kNN	0.258	0.271	0.270	0.242	0.258	0.260	0.257	0.248	0.357	0.355	0.360	0.319
	NB	0.242	0.243	0.245	0.236	0.274	0.274	0.287	0.263	0.275	0.288	0.296	0.284
	SVM	0.264	0.264	0.264	0.245	0.298	0.298	0.298	0.287	0.264	0.264	0.264	0.293
	C4.5	0.260	0.314	0.307	0.281	0.287	0.321	0.308	0.319	0.289	0.307	0.274	0.254
210	kNN	0.233	0.233	0.233	0.188	0.269	0.267	0.276	0.253	0.352	0.352	0.352	0.306
	NB	0.236	0.248	0.243	0.206	0.264	0.267	0.286	0.294	0.307	0.281	0.295	0.265
	SVM	0.238	0.238	0.238	0.188	0.279	0.262	0.262	0.235	0.233	0.233	0.371	0.282
	C4.5	0.281	0.348	0.331	0.259	0.319	0.367	0.283	0.312	0.319	0.291	0.324	0.282
105	kNN	0.324	0.324	0.324	0.282	0.271	0.267	0.267	0.235	0.400	0.391	0.391	0.306
	NB	0.224	0.210	0.229	0.188	0.348	0.305	0.381	0.188	0.252	0.295	0.267	0.188
	SVM	0.229	0.229	0.229	0.212	0.257	0.257	0.257	0.235	0.276	0.276	0.276	0.235
	C4.5	0.314	0.314	0.305	0.259	0.362	0.362	0.343	0.188	0.343	0.343	0.295	0.282
53	kNN	0.286	0.286	0.286	0.150	0.267	0.267	0.267	0.200	0.210	0.210	0.210	0.150
	NB	0.324	0.362	0.343	0.250	0.276	0.286	0.305	0.200	0.229	0.229	0.248	0.150
	SVM	0.343	0.343	0.343	0.300	0.267	0.267	0.267	0.200	0.210	0.210	0.210	0.150
	C4.5	0.343	0.324	0.267	0.100	0.352	0.362	0.352	0.200	0.171	0.191	0.191	0.100

It is easy to notice that for datasets of bigger number of instances (from 1676 to 419) the best results for different classifiers were obtained for various methods. And the optimal technique cannot be indicated. However, for the smaller datasets of 210, 105 and 53 instances, *Labels Chain (LC)* performs the best. In the 16 out of 36 cases *Hamming Loss* values were even less or equal to 0.200.

During experiments more strict measure *Classification Accuracy* was considered. In that case, the trend of shadowed best results is similar to the one of *Hamming Loss*. The obtained results for bigger datasets have no repeatability for different methods, while for smaller ones best values of *Classification Accuracy* were provided for *Labels Chain* algorithm.

Table 2. Datasets with 2 labels assigned – results of Classification Accuracy [%]

Instances, classifiers		TwoLabels_1				TwoLabels_2				TwoLabels_3			
		BR	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**
1676	kNN	46.42	46.87	47.01	36.19	32.69	36.27	34.48	29.85	41.34	42.09	41.94	39.93
	NB	35.22	46.12	45.52	39.18	33.88	46.72	44.03	42.16	36.12	51.04	44.48	46.64
	SVM	36.57	36.57	36.57	36.94	36.87	36.87	36.87	37.69	37.01	37.01	37.16	38.06
	C4.5	28.36	41.94	40.75	32.46	32.69	40.00	35.67	34.33	28.81	38.81	37.91	36.19
838	kNN	26.73	27.03	27.33	31.58	33.43	34.63	34.63	38.81	33.13	35.82	34.03	30.60
	NB	31.53	39.64	35.44	31.58	32.54	48.06	41.49	39.55	31.04	46.27	41.79	39.55
	SVM	38.44	38.44	38.44	40.60	36.42	36.42	36.42	36.57	33.73	33.73	33.73	41.04
	C4.5	26.43	37.24	31.83	25.56	27.46	32.84	35.82	34.33	23.88	35.22	33.13	27.61
419	kNN	39.88	37.50	37.50	41.79	38.69	39.29	39.88	38.81	20.24	23.21	20.83	23.88
	NB	33.93	45.24	42.26	43.28	27.38	37.50	33.33	38.81	25.00	35.12	31.55	38.81
	SVM	39.29	39.29	39.29	40.30	35.71	35.71	35.71	37.31	40.48	40.48	40.48	34.33
	C4.5	22.02	30.95	31.55	34.33	16.67	30.95	29.17	31.34	20.83	28.57	34.52	40.30
210	kNN	46.43	46.43	46.43	52.94	41.67	44.05	42.86	44.12	17.86	17.86	17.86	32.35
	NB	36.90	44.05	44.05	50.00	30.95	35.71	36.90	29.41	16.67	34.52	32.14	41.18
	SVM	45.24	45.24	45.24	52.94	27.38	41.67	41.67	47.06	47.62	47.62	15.48	38.24
	C4.5	16.67	26.19	26.19	38.24	19.05	20.24	28.57	29.41	16.67	33.33	26.19	41.18
105	kNN	21.43	21.43	21.43	29.41	38.10	38.10	38.10	41.18	19.05	21.43	21.43	23.53
	NB	45.24	57.14	45.24	52.94	16.67	30.95	19.05	52.94	35.71	33.33	38.10	47.06
	SVM	50.00	50.00	50.00	52.94	40.48	40.48	40.48	41.18	38.10	38.10	38.10	47.06
	C4.5	21.43	30.95	28.57	41.18	14.29	21.43	19.05	52.94	16.67	21.43	33.33	41.18
53	kNN	42.86	42.86	42.86	62.50	42.86	42.86	42.86	50.00	47.62	47.62	47.62	62.50
	NB	28.57	23.81	28.57	37.50	33.33	38.10	33.33	62.50	38.10	42.86	38.10	62.50
	SVM	28.57	28.57	28.57	37.50	42.86	42.86	42.86	50.00	47.62	47.62	47.62	62.50
	C4.5	23.81	38.10	52.38	75.00	0.00	23.81	14.29	50.00	52.38	52.38	52.38	75.00

Similarly to 2 label datasets, the experiments were carried out on datasets with 3 labels (*ThreeLabels_1*, *ThreeLabels_2* and *ThreeLabels_3*). Results of *Hamming Loss* measure are presented in Tab. 3. Overview of the table shows tendency similar to the first part of the experiments. Only with smaller datasets the observed trend stabilizes and the best results are almost always obtained for *Labels Chain* method. There are only 3 exceptions for *ThreeLabels_1* dataset – SVM and C4.5 for 150 instances and SVM for 38 ones.

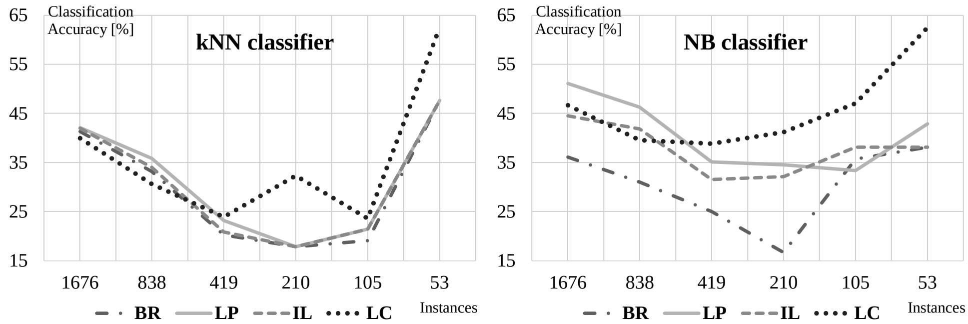

Figure 1. Dataset *TwoLabels_3* – comparison of Classification Accuracy results for kNN and NB classifiers

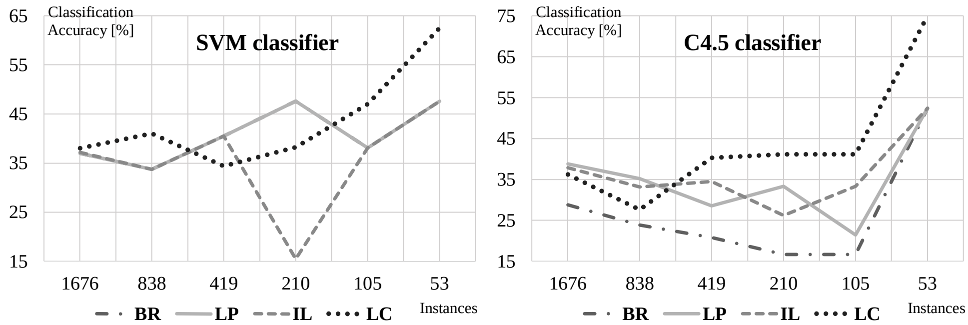

Figure 2. Dataset *TwoLabels_3* – comparison of Classification Accuracy results for SVM and C4.5 classifiers

As it can be easy noticed in Tab. 4, Fig. 3 and Fig. 4, *Classification Accuracy* results confirm the effectiveness of the considered method for the dataset of the smallest sizes. *Labels Chain* algorithm gave the best results for *ThreaLabels_2* with 300, 150, 75 and 38 objects, and for *ThreaLabels_1* and *ThreeLabels_3* with 150, 75 and 38 objects. The exceptions occurred only for *ThreaLabels_1* with NB for 150 objects and SVM for 38 ones.

Summing up, during the experiments, there is observed the similar trend in obtained results for *Hamming Loss* measure as well as *Classification Accuracy*. The *Labels Chain* method achieved the best results for all the classifiers for datasets of small number of instances. It is also worth to mention that *LC* method gave much better results than its basic version *Independent Labels*. Thus, one can conclude that mapping dependencies between labels should ameliorate multi-label classification performance.

Table 3. Datasets with 3 labels assigned – results of Hamming Loss

Instances, classifiers		ThreeLabels_1				ThreeLabels_2				ThreeLabels_3			
		BR	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**
1200	kNN	0.266	0.267	0.290	0.197	0.328	0.325	0.314	0.257	0.218	0.218	0.282	0.530
	NB	0.198	0.172	0.273	0.177	0.196	0.178	0.284	0.203	0.217	0.203	0.303	0.239
	SVM	0.208	0.208	0.285	0.218	0.208	0.208	0.293	0.197	0.219	0.219	0.301	0.177
	C4.5	0.203	0.226	0.284	0.192	0.214	0.238	0.286	0.260	0.222	0.210	0.308	0.213
600	kNN	0.278	0.270	0.307	0.211	0.290	0.293	0.302	0.247	0.238	0.242	0.282	0.232
	NB	0.190	0.170	0.287	0.153	0.208	0.207	0.281	0.258	0.203	0.197	0.287	0.200
	SVM	0.190	0.190	0.282	0.158	0.227	0.227	0.287	0.242	0.212	0.212	0.290	0.189
	C4.5	0.211	0.215	0.302	0.201	0.248	0.228	0.315	0.179	0.209	0.213	0.306	0.226
300	kNN	0.315	0.313	0.323	0.274	0.233	0.237	0.317	0.168	0.237	0.233	0.297	0.189
	NB	0.228	0.220	0.313	0.200	0.207	0.210	0.300	0.189	0.207	0.203	0.293	0.211
	SVM	0.230	0.230	0.307	0.316	0.233	0.233	0.303	0.147	0.197	0.197	0.290	0.211
	C4.5	0.262	0.263	0.327	0.200	0.232	0.260	0.323	0.232	0.245	0.307	0.337	0.211
150	kNN	0.270	0.280	0.267	0.200	0.230	0.227	0.293	0.160	0.367	0.360	0.313	0.200
	NB	0.233	0.220	0.280	0.200	0.187	0.193	0.313	0.160	0.177	0.187	0.293	0.160
	SVM	0.227	0.227	0.287	0.240	0.220	0.220	0.300	0.120	0.160	0.160	0.300	0.160
	C4.5	0.257	0.293	0.283	0.280	0.210	0.253	0.323	0.200	0.250	0.260	0.327	0.120
75	kNN	0.440	0.440	0.493	0.240	0.207	0.213	0.280	0.080	0.427	0.427	0.400	0.280
	NB	0.260	0.267	0.307	0.160	0.193	0.187	0.280	0.160	0.180	0.173	0.253	0.080
	SVM	0.267	0.267	0.307	0.240	0.227	0.200	0.267	0.160	0.173	0.173	0.253	0.160
	C4.5	0.287	0.280	0.307	0.240	0.360	0.240	0.313	0.080	0.340	0.267	0.327	0.080
38	kNN	0.333	0.347	0.280	0.200	0.293	0.293	0.360	0.100	0.320	0.320	0.387	0.100
	NB	0.200	0.213	0.307	0.000	0.320	0.267	0.360	0.200	0.187	0.187	0.333	0.000
	SVM	0.187	0.187	0.307	0.200	0.320	0.320	0.360	0.200	0.187	0.187	0.333	0.000
	C4.5	0.280	0.400	0.360	0.000	0.360	0.293	0.373	0.200	0.227	0.267	0.360	0.000

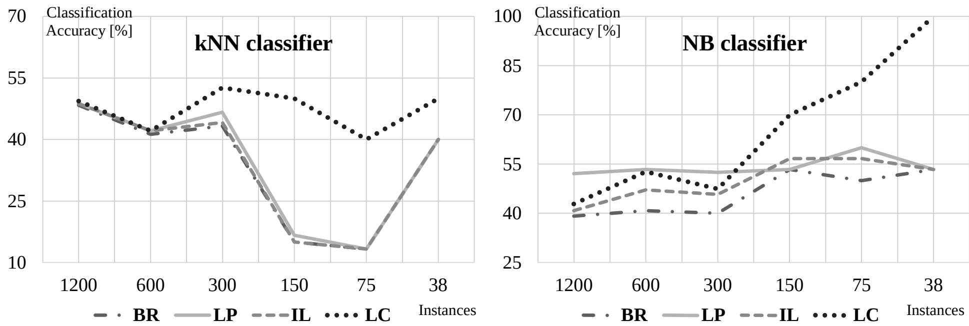

Figure 3. Dataset *ThreeLabels_3* – comparison of Classification Accuracy results for kNN and NB classifiers

Table 4. Datasets with 3 labels assigned – results of Classification Accuracy [%]

Instances, classifiers		ThreeLabels_1				ThreeLabels_2				ThreeLabels_3			
		BR	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**	**BR**	**LP**	**IL**	**LC**
1200	kNN	41.67	41.88	41.67	50.65	27.71	31.25	29.58	37.66	48.33	48.75	48.75	49.35
	NB	41.04	57.71	47.29	57.14	42.71	56.88	48.33	48.05	39.17	52.08	40.83	42.86
	SVM	51.67	51.67	55.21	49.35	50.83	50.83	33.54	53.25	47.92	47.92	48.96	57.14
	C4.5	35.42	46.88	47.29	55.84	33.33	45.00	43.33	36.36	33.13	52.08	45.42	51.95
600	kNN	37.92	40.42	39.17	50.00	32.92	32.92	32.92	39.47	41.25	42.08	42.08	42.11
	NB	45.00	59.58	48.75	60.53	40.42	52.08	46.25	42.11	40.83	53.33	47.08	52.63
	SVM	55.42	55.42	55.42	60.53	46.25	46.25	46.25	44.74	51.25	51.25	51.25	52.63
	C4.5	31.67	49.17	40.83	47.37	27.50	46.67	37.92	52.63	40.42	50.42	38.33	47.37
300	kNN	27.50	28.33	27.50	31.58	40.83	43.33	41.67	63.16	43.33	46.67	44.17	52.63
	NB	37.50	49.17	41.67	52.63	37.50	50.83	43.33	57.89	40.00	52.50	45.83	47.37
	SVM	48.33	48.33	48.33	21.05	44.17	44.17	44.17	63.16	53.33	53.33	53.33	52.63
	C4.5	26.67	40.00	37.50	57.89	28.33	39.17	39.17	42.11	20.83	32.50	30.00	47.37
150	kNN	33.33	35.00	36.67	60.00	45.00	46.67	46.67	50.00	15.00	16.67	15.00	50.00
	NB	36.67	53.33	46.67	50.00	45.00	51.67	43.33	60.00	53.33	53.33	56.67	70.00
	SVM	50.00	50.00	50.00	50.00	46.67	46.67	46.67	70.00	60.00	60.00	60.00	60.00
	C4.5	30.00	33.33	38.33	40.00	38.33	38.33	43.33	50.00	26.67	41.67	26.67	60.00
75	kNN	23.33	23.33	23.33	40.00	46.67	46.67	46.67	80.00	13.33	13.33	13.33	40.00
	NB	30.00	43.33	36.67	80.00	36.67	53.33	40.00	60.00	50.00	60.00	56.67	80.00
	SVM	43.33	43.33	43.33	60.00	0.00	50.00	26.67	60.00	60.00	60.00	60.00	60.00
	C4.5	30.00	40.00	36.67	60.00	16.67	46.67	36.67	60.00	10.00	43.33	30.00	80.00
38	kNN	13.33	13.33	13.33	50.00	33.33	33.33	33.33	50.00	40.00	40.00	40.00	50.00
	NB	40.00	46.67	40.00	100.00	33.33	40.00	33.33	50.00	53.33	53.33	53.33	100.00
	SVM	60.00	60.00	60.00	50.00	33.33	33.33	33.33	50.00	53.33	53.33	53.33	100.00
	C4.5	13.33	20.00	20.00	100.00	6.67	40.00	20.00	50.00	33.33	46.67	46.67	100.00

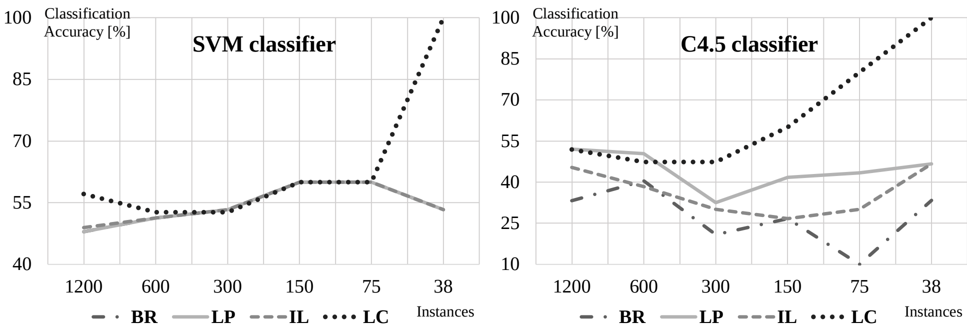

Figure 4. Dataset *ThreeLabels_3* – comparison of Classification Accuracy results for SVM and C4.5 classifiers

5. Conclusion

In the paper, new effective problem transformation method of multi-label classification for text document datasets is presented. The experiments carried out on datasets of different number of attributes and different sizes showed the good performance of the proposed *Labels Chain* method, comparing to the problem transformation methods like *Binary Relevance* and *Label Power-set*. Especially, the best results were obtained for datasets of big number of attributes and relatively small number of instances.

Future investigations will consist in conducting further experiments taking into account text datasets of different sizes and different number of attributes. It is also worth considering to examine the performance of the method taking into account bigger number of relevant labels, as well as using different evaluation criteria.

REFERENCES

[1] Glinka K., Zakrzewska D. (2015) *Effective Multi-label Classification Method for Multidimensional Datasets*, Proceeding of the 11th International Conference FQAS 2015, Cracow, Poland, 127–138.

[2] Schapire R.E., Singer Y. (2000) *BoosTexter: A boosting-based system for text categorization*, Machine learning 39(2/3), 135-168.

[3] Li T., Ogihara M. (2004) *Content-based music similarity search and emotion detection,* Proceeding of IEEE International Conference on Acoustic, Speech and Signal Processing (volume 5), Canada, 705–708.

[4] Tsoumakas G., Katakis I., Vlahavas I. (2010) *Mining Multi-label Data*, Maimon O., Rokach L. [ed.]: Data Mining and Knowledge Discovery Handbook, Springer US, Boston, MA, 667-685.

[5] Madjarov G., Kocev D., Gjorgjevikj D., Dẑeroski S. (2012) *An extensive experimental comparison of methods for multi-label learning*, Pattern Recognition 45(9), 3084-3104.

[6] Sajnani H., Javanmardi S., McDonald D.W., Lopes C.V. (2011) *Multi-label classification of short text: A study on wikipedia barnstars*, Analyzing Microtext: Papers from the 2011 AAAI Workshop.

[7] Boutell M.R., Luo J., Shen X., Brown C.M. (2004) *Learning multi-label scene classification,* Pattern Recognition 37(9), 1757–1771.

[8] Esuli A., Fagni T., Sebastiani F. (2008) *Boosting multi-label hierarchical text categorization,* Information Retrieval 11(4), 287-313.

[9] Comité F.D., Gilleron R., Tommasi M. (2003) *Learning multi-label alternating decision decision tree from text and data*, Lecture Notes in Computer Science, vol. 2734, Springer, Heidelberg, 35-49.

[10] Lee S.-J., Jiang J.-Y. (2014) *Multilabel text categorization based on fuzzy relevance clustering*, IEEE Transactions on Fuzzy Systems 22(6), 1457-1471.

[11] Read J., Pfahringer B., Holmes G., Frank E. (2009) *Classifier Chains for Multi-label Classification,* Buntine W., Grobelnik M., Mladenic, D., Shawe-Taylor J. [ed.]: Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science, vol. 5782, Springer, Heidelberg, 254–269.

[12] Kajdanowicz T., Kazienko P. (2012) *Multi-label classification using error correcting out-put codes*, Applied Mathematics and Computer Science 22(4), 829–840.

[13] http://www.yelp.com/

[14] http://www.ics.uci.edu/~vpsaini/

[15] Koehn P. (2010) *Statistical Machine Translator*, Cambridge University Press, UK.

[16] Witten I.H., Frank E., Hall M.A. (2011) *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, San Francisco, USA.

[17] http://www.cs.waikato.ac.nz/ml/weka/index.html