A Diversified Classification Committee for Recognition of Innovative Internet Domains

Marcin Mirończuk(✉) and Jarosław Protasiewicz

Laboratory of Intelligent Information Systems,
National Information Processing Institute, al. Niepodległości 188b,
00-608 Warsaw, Poland
marcin.mironczuk@opi.org.pl
http://lis.opi.org.pl

Abstract. The objective of this paper was to propose a classification method of innovative domains on the Internet. The proposed approach helped to estimate whether companies are innovative or not through analyzing their web pages. A Naïve Bayes classification committee was used as the classification system of the domains. The classifiers in the committee were based concurrently on Bernoulli and Multinomial feature distribution models, which were selected depending on the diversity of input data. Moreover, the information retrieval procedures were applied to find such documents in domains that most likely indicate innovativeness. The proposed methods have been verified experimentally. The results have shown that the diversified classification committee combined with the information retrieval approach in the preprocessing phase boosts the classification quality of domains that may represent innovative companies. This approach may be applied to other classification tasks.

Keywords: Text mining · Classification · Text classification · Information retrieval · Committee classification

1 Introduction

It is believed that innovativeness plays an important role in the development of modern economies. Since it depends on cooperation between companies and researchers, we have decided to create an information platform called Inventorum, which is aimed to boost information flow between these two sides. The platform is a recommender system that proposes innovations, projects, partners and experts that suit both, companies and research teams best. Among many processes that this system contains, there is one especially important, i.e. finding potentially innovative companies on the Internet, which should help to find more participants of the Inventorum. The system collects data from the Internet. Then each acquired domain (a group of web pages) is analyzed in order to find out whether it is related to an innovative company. This study deals with the classification issue of such domains into innovative companies and the others.

S. Kozielski et al. (Eds.): BDAS 2016, CCIS 613, pp. 368–383, 2016.
DOI: 10.1007/978-3-319-34099-9_29

The term "an innovative domain" is an abstract idea similar to such concepts like spam, pornography, or sport, etc. These domains may be automatically recognized on the Internet by supervised machine learning methods. Unfortunately, the innovativeness is hard to define; however, the research conducted by Leon Kozminski Academy provided some definitions and depicted attributes that may characterize an innovative company. The research was based on questionnaires that covered several useful indicators of innovativeness; thus, they helped to classify companies. However, it is almost impossible to retrieve these indicators from company's domains automatically, and then create the profiles of considered businesses. Beside directly built models of innovativeness, there are annual rankings of leading companies on the market, e.g. Forbes rankings, awards like Business Gazelle, etc. We assumed that these rankings are highly reliable and decided to utilize these data to verify our hypothesis as follows. We assume that it is possible to create a classification model that can decide whether a company is innovative or not based on automatic analysis of its Internet domain.

The review of the recent literature showed that there are some works [9–11,15] concerning the detection of innovative themes on the Internet; however, the analyzed approaches are inappropriate to solve sufficiently the problem posed in our article. Of course, there are a plenty of works concerning the problem of documents classification in various areas like spam detection, porn sites recognition, security issues, or medical documentation analysis. Nonetheless, these solutions are designed for limited purposes and are unsuitable for the problem of innovative sites detection. Thus, we experimentally constructed a classification model that can recognize innovative domains on the Internet with sufficient quality.

In this study, we use some text mining techniques. The text mining is related to data mining. David Hand et al. [5] defined data mining as an analysis of observable and often large data sets to find unsuspected relationships and to summarize data in novel ways that will provide understandable and useful information to a data owner. Text mining relates to data mining methodologies applied to textual sources. It covers various approaches to text analysis, which mainly are [16,19] classification, clustering, translation, information retrieval, and summarization or information extraction.

The main aim of this work is to classify domains collected from the Internet to *innovative* or *no innovative* groups based on the documents that they include. To resolve this task, we propose an experimental Naïve Bayes (NB) classification committee. The committee is based on a diversified feature space, feature weights and two models of feature distribution, i.e. Bernoulli and Multinomial models. In the experiments, we verify whether this committee is enough diversified to resolve the classification task mentioned above and if the committee improve the classification quality in comparison to a single NB classifier. We provide the results of the whole classification committee as well as the performance of the individual NB classifiers composing the committee. Thus, it is easy to notice advances of using the committee in comparison to single classification models.

Furthermore, the presented study is theoretically well grounded to give a deep understanding of the proposed methods as well as to be easy to reproduce.

It is worth to note that as a result of several experiments, we elaborated the classification model consisting a classifiers committee. This non-trivial model covers advanced methods of features construction. We have to underline that the various single classifiers (k-NN, decision trees, support vector machines, etc.) have been verified experimentally prior to construction of the final model. Moreover, the experiments involved dimensionality reduction of the feature space by using typical methods like principal component analysis, singular value decomposition, and filters. Unfortunately, the examined classifiers produced such poor decisions that we decided to exclude their results from the article. Since the use of a single classifier turned out to be inappropriate to the task under examination, we worked out the more complicated and sophisticated solution that finally gave the sufficient results. The model was tested on a new and unique test set constructed by us only for the purpose of this study.

The paper is structured as follows. Section 2 contains the definitions and mathematical background of text mining techniques and components, which are used to resolve the defined classification problem. Section 3 presents an overview of the proposed innovative domains classification system. Next, Sect. 4 describes the evaluation process of the proposed system and the results obtained during experiments. Finally, Sect. 5 concludes the findings.

2 Text Mining and Mathematical Background

This section presents all necessary definitions and mathematical background that are used in the proposed classification system. Definition 1 presented below explains the term *text mining* [16].

Definition 1. *The term "text mining" is used analogously to data mining when data are text. As there are some data specificities when handling text compared to handling data from databases, text mining has some specific methods and approaches. Some of these are extensions of data mining and machine learning methods while other are rather text-specific. Text mining approaches combine methods from several related fields, including machine learning, data mining, information retrieval, natural language processing, statistical learning, and the Semantic Web.*

Usually, we build a processing pipeline to process a text. Figure 1 presents the basic pipeline of text processing.

Text corpora → Feature construction → Text representation →
Feature selection → Text mining tasks

Fig. 1. The basic pipeline of text processing.

A *text corpora* (*a corpus of texts* or shortly *a corpus*) D is a collection of documents $d \in D$. Solka [19] defined a *document* as a sequence of words and punctuations following the grammatical rules of language. A document is any relevant segment of a text and can be of any length. Examples of documents include sentences, paragraphs, sections, chapters, books, web pages, emails, etc. A *term* (feature) is usually a word (uni-gram), but it can also be a word-pair (bi-gram) or a phrase (n-gram). Terms are constructed by a *Feature construction* component, and they build an appropriate *Text representation* (Subsect. 2.1). Usually, when the *Text representation* is created we try to reduce the number of terms using a *Feature selection* component. Subsect. 2.2 presents several feature selection methods. Generally, after the *Feature selection* phase follows the realization of an appropriate *Text mining task*. In our solution, we applied two *Text mining tasks*. The first task is related to *Information retrieval*, which also supports a realization of the second task, i.e. classification. Subsect. 2.3 briefly describes the information retrieval approach and presents the Okapi BM25 ranking function used by search engines. We use this ranking function to find the best innovative web documents per domain and build a classification model. The proposed classification process is based on the Naïve Bayes classifier, which is described in Subsect. 2.4.

2.1 Feature Construction and Text Representation

Definition 2 presented below explains the term *feature construction* in the context of text mining [16].

Definition 2. *Feature construction in text mining consists of various techniques and approaches, which convert textual data into a feature-based representation. Since traditional machine learning and data mining techniques are generally not designed to deal directly with textual data, feature construction is an important preliminary step in text mining converting source documents into a representation that a data mining algorithm can then work with it. Various kinds of feature construction approaches are used in text mining depending on a task that is being addressed or data mining algorithms and the nature of the dataset in question.*

Usually, the document preprocessing methods are used in the *feature construction* task. These methods use *natural language processing* (NLP) techniques. Figure 2 presents the basic pipeline of document pre-processing methods.

Document → Tokenization → End sentence recognition →
Morphological analysis → Morphological disambiguation → Labeling

Fig. 2. The basic pipeline of document pre-processing methods.

As shown in Fig. 2, each *document* from a *Text corpora* is tokenized. The *tokenization* process splits all documents into words. A *word* is a sequence of letters in a text written in a natural language and usually separated by either spaces or punctuation marks [21]. Usually, after the *tokenization* we need to know where is the end of a sentence. It is recognized by the *End sentence recognition* process. After that, each sentence is processed by the *Morphological analysis* component. The *morphological analysis* relies on the determination of all morphological forms of all particular words. For each word, we try to find all *lemmas* (base forms) and all tags. A tag contains values of grammatical categories specifying the form. The *morphological disambiguation* determine the form realized by a particular occurrence of a word in its context [21]. The sequence of a morphological analysis and disambiguation is in jargon referred as *tagging* [21]. The *labeling* component is used to create the final features set. In the simplest way, we may take to the analysis (to resolve *Text mining tasks*) all *words* after *tokenization* process. We can reduce this set by using a *stop list*. The *stop list* contains words/features that we remove from document, i.e. we remove from *document* all occurrences of each word/feature from the stop list. Also, we can use some heuristics methods (manual methods) to create the set of features. Moreover, we can create own ontology or use a more sophisticated ontology like *Słowosieć* (a Polish *WordNet*) [12], for example, to unify words (to find one basic synonym of words). Also, we may use a more sophisticated feature construction based on a statistical word co-occurrence analysis [4] or NLP shallow parsing technique [13]. In the first case, we create the set of features (n-grams) by finding a *frequent occurrence* of $2, 3, ..., n$ words like a *business intelligence* or *commercial enterprise*. In the second case, we create the set of features by finding, for example, noun or verb phrases like an *innovative technology* or a *fast car*, etc.

Table 1 presents the results, i.e. an example of the *tagging* process. We considered the document example d – *"Mam próbkę analizy morfologicznej."* (*"I have a morphological analysis sample."*)

Table 1. The example of the tagging process. Source [1].

0	Mam	mama [mother]	fin:sg:ter:subst:pl:gen:f
		mamić [to beguile]	impt:sg:sec:imperf
		mieć [to have]	fin:sg:pri:imperf
1	próbkę	próbka [sample]	subst:sg:acc:f
2	analizy	analiza [analysis]	subst:sg:gen:f
			subst:pl:nom.acc.voc:f
3	morfologicznej	morfologiczny [morphological]	adj:sg:gen.dat.loc:f:pos
4	.	.	interp

According to [1], we can consider each line of Table 1 as one morphological interpretation, the horizontal lines separate the groups of analysis of particular words. The input document was segmented into words (particularly the full stop was separated from the word "morfologicznej"). On the right, corresponding lemmas (entries) are provided. The next column contains tags describing values of grammatical categories (IPI PAN morphological tagset [20]) of particular forms.

After the *Morphological disambiguation* phase, and when we take only *lemmas*, is received the following document $d' = \{\{$*mieć,[fin:sg:pri:imperf]*$\}$, $\{$*próbka, [subst:sg:acc:f]*$\}$, $\{$*analiza, [subst:sg:gen:f]*$\}$, $\{$*morfologiczny [adj:sg:gen.dat.loc: f:pos]*$\}$, $\{$*., [interp]*$\}\}$. Based on this document and the *Labeling* techniques mentioned above, we can construct the following example set of features: $s = \{$*mieć, próbka, analiza, morfologiczny*$\}$ - this set will be created when we will use *stop words set* like a $S_w = \{.\}$; $s = \{$*próba, analiza, morfologiczny*$\}$ - this set will be created when we will use S_w $=\{$.,mieć$\}$ and when we will replace feature *próbka* by its synonym, i.e. *próba*; $s = \{$*próbka-analiza-morfologiczny, analiza-morfologiczny*$\}$ - this set will be created when we will use NLP shallow parsing technique to recognition *noun phrase* NP; $s = \{$*próbka-analiza, analiza-morfologiczny*$\}$ - this set will be created when we will use the co-occurrence recognition techniques; $s = \{$*fin, subst, adj, interp, NP, NP*$\}$ - this set will be created when we will use for example only the *part of speech* and labels of the recognized NP; s as the combination of the mentioned above sets.

After the *feature construction* phase, we may construct an appropriate text representation. There are two main representations of document [5,8,17]: a graph or a vector space model VSM (a document-term matrix or a term-document matrix). Schenker and all [17] modeled the document as a graph, where features are vertexes, and edges model connections between features. The VSM approach represents the collection of documents as a matrix.

The document-term matrix has n $(1 \leq i \leq n)$ rows and m $(1 \leq j \leq m)$ columns, where m represents the number of features in the corpus D, and n is the number of documents. The w_{ij} element of the matrix D relates to the weight of the term f_j in the document d_i. There are a few typical weighting functions such as [8] binary, term-frequency (TF), invert document frequency (IDF) or mixed $TF \times IDF$, etc. In this study, we use binary and TF weighting functions. The binary weighting function sets the weight w_{ij} equal to 1 $(w_{ij} = 1)$ if and only if the feature f_j occurs in the document d_i. If the feature f_j does not occur in the document d_i, the weight w_{ij} is set to 0 $(w_{ij} = 0)$. The TF weighting function counts the number of times the j-th feature appears in the i-th document. Encoding the corpus as the matrix allows to utilize the power of linear algebra and quickly analyze the documents collection. All presented solutions like the feature selection, the ranking functions used by search engines or classification methods are based on the VSM model discussed above.

2.2 Feature Selection Methods and Filter Methods

Definition 3 presented below explains the term *feature selection* in the context of text mining [16].

Definition 3. *The term "feature selection" is used in machine learning for the process of selecting a subset of features (dimensions) used to represent the data. Feature selection can be seen as a part of data pre-processing potentially followed or coupled with feature construction, but can also be coupled with the learning phase if embedded in the learning algorithm. An assumption of feature selection* *is that we have defined an original feature space that can be used to represent the data, and our goal is to reduce its dimensionality by selecting a subset of original features.*

We can divide the feature selection into three main groups [2], namely filters, wrappers, and embedded methods. In this study, we use the filter methods. These methods utilize a feature ranking function to choose the best features. The ranking function gives a relevance score based on a sequence of examples. Intuitively, more relevant features will be higher in the rank. Thus, we may keep n-top features or remove n-worst ranked features from the dataset. These methods are often univariate and consider each feature independently or with regard to a dependent variable. For example, some filter methods include the χ^2 squared test, information gain, and correlation coefficient scores. In the presented research the following filter methods are used [3,6,8,18]: fisher ranking (Fisher), correlation coefficients (Cor), mutual information (MI), normalized punctual mutual information (NPMI), χ^2 (Chi), Kolmogorov–Smirnov test (KS), and Mann–Whitney U test (WC).

2.3 Information Retrieval and The Okapi BM25 Search Engine

Definition 4 presented below explains the term *information retrieval* in the context of text mining [8,16].

Definition 4. *Information retrieval (IR) is a set of techniques that extract from a collection of documents those that are relevant to a given query. Initially addressing the needs of librarians and specialists, the field has evolved dramatically with the advent of the World Wide Web. It is more general than data retrieval, which purpose is to determine which documents contain occurrences of the keywords that make up a query. Whereas the syntax and semantics of data retrieval frameworks are strictly defined, with queries expressed in a totally formalized language, words from a natural language give no or limited structure are the medium of communication for information retrieval frameworks. A crucial task for an IR system is to index the collection of documents to make their contents efficiently accessible. The documents retrieved by the system are usually ranked by expected relevance, and the user who examines some of them might be able to provide feedback so that the query can be reformulated and the results improved.*

The Okapi BM25 retrieval function is state-of-the-art of information retrieval systems [8,14]. It was first implemented in London's City University in the 1980s and 1990s by Stephen E. Robertson, Karen Spärck Jones, and others. The Okapi BM25 is a bag-of-words retrieval function that is used by search engines to rank matching documents d according to their relevance to a given search query q. It is not a single function, but the whole family of scoring functions with slightly different components and parameters. One of the most popular instantiations of this function is the BM25 scoring function that is defined by Eq. 1.

$$scoring(q, d) = \sum_{i=1}^{|q|} idf(q_i) \cdot \frac{tf(q_i, d) \cdot (k_1 + 1)}{tf(q_i, d) + k_1 \cdot (1 - b + b \cdot \frac{|d|}{avg_{docs}})} \tag{1}$$

where: q is the query that consists of features/terms; d is the document (the bag of features/terms/words); $f(q_i, d)$ correlates to the term's frequency defined as the number of times that query term q_i appears in the document d; d is the length of the document d in words; avg_{docs} is the average document length over all documents in the collection; k_1 and b are free parameters usually equal to $k_1 = 2.0$ and $b = 0.75$; $idf(q_i)$ is the inverse document frequency weight of the query term q_i.

The Eq. 2 describes how $idf(q_i)$ is computed.

$$idf(q_i) = log(\frac{N - df(q_i) + 0.5}{df(q_i) + 0.5}) \tag{2}$$

where: N is the total number of documents in the collection; $df(q_i)$ is the number of documents containing the query term q_i.

2.4 Text Classification and the Naïve Bayes classifier

Definition 5 presented below explains the term *text classification* in the context of text mining [7,8].

Definition 5. *Text classification or categorization is the problem of learning classification models from training documents labeled by pre-defined classes. Then, such models are used to classify new documents. For example, we have a set of web page documents that belongs to two classes or topics, e.g. companies and no-companies. We want to learn a classifier that is able to classify new documents into these classes.*

We can compute the probability P of the document d belonging to a class c thanks to Naïve Bayes Theorem. Equation 3 shows how we can compute this probability.

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)} \propto P(c)P(d|c) = P(c) \prod_{1 \leq k \leq n_d} P(f_k|c) \tag{3}$$

where: $P(f_k|c)$ is the conditional probability of the feature f_k occurring in the document of the class $c \in C$; $P(c)$ is the prior probability of the document occurring in class c; n_d is the number of features f in the document d.

In the text classification task, our goal is to find the *best* class c for the document d. The best class in NB classification is the most likely or *maximum a posteriori* (MAP) class c_{map}. Equation 4 presents c_{map} for the Bernoulli model feature distribution, and Eq. 7 presents c_{map} for the Multinomial model feature distribution.

$$c_{map,Bernoulli} = \arg\max_{c\in C}[\log \hat{P}(c) + \sum_{1\leq k\leq n_d} (\log(b_{f_k}\hat{P}(f_k|c)) + \log((1-b_{f_k})(1-\hat{P}(f_k|c)))] \quad (4)$$

where: b_{f_k} - $b_{f_k} = 1$ if the feature f_k is present in the document d, otherwise $b_{f_k} = 0$.

We can estimate $\hat{P}(c)$ and $\hat{P}(f_k|c)$ by using Eqs. 5 and 6 respectively.

$$\hat{P}(c) = \frac{N_c}{N} \quad (5)$$

where: N_c is the number of documents in the class c; N is the total number of documents.

$$\hat{P}(f_k|c) = \frac{N_{c,f}+1}{N_c+2} \quad (6)$$

where: $N_{c,f}$ is the number of document in the class c that contain the feature f; N_c see Eq. 5.

$$c_{map,Multinomial} = \arg\max_{c\in C}[\log \hat{P}(c) + \sum_{1\leq k\leq n_d} \log(\hat{P}(f_k|c))] \quad (7)$$

where: $\hat{P}(f_k|c)$ we can be estimated by Eq. 8.

$$\hat{P}(f_k|c) = \frac{T_{c,f}+1}{\sum_{f^{\text{‘}}\in F}(T_{c,f^{\text{‘}}}+1)} \quad (8)$$

where: F is the set of features (vocabulary) of the text corpora; $T_{c,f}$ is the number of occurrences of the feature f in training documents belonging to the class c including multiple occurrences of this feature in the document.

3 Innovative Domains Classification System - Overview

This section describes the proposed classification system of innovative domains. Figure 3 presents a basic flow and components of the system.

A component for *Crawling of the potentially innovative domains* (see Fig. 3), based on the initial list of companies (domains), is used for browsing the World Wide Web (Internet) and download its content for each domain, i.e. all domains of web pages. All downloaded pages are stored in *The potential innovative domains of companies and their documents* data-store D_{all}. After that, a *Recognition of the innovative logo for each domain* component recognizes if a domain contains any innovative logo. The innovative logo is a logotype such as “gazela biznesu”, “diamenty forbesa”, Europe Union logotypes, etc. All results of this recognition are stored in *The innovative logo of companies* data-store D_{logo}.

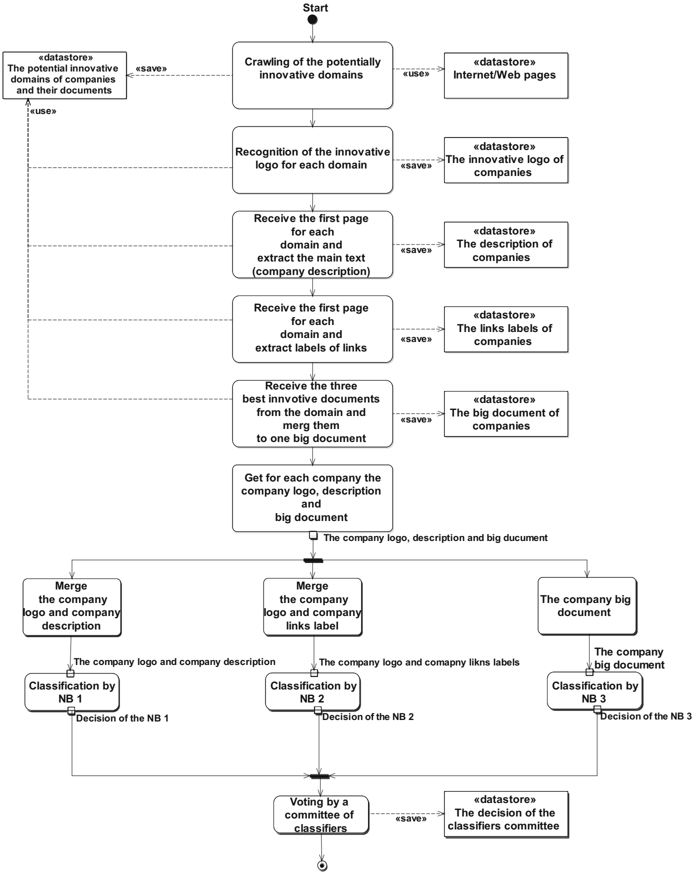
Start
«datastore»
The potential innovative domains of companies and their documents
«save»
Crawling of the potentially innovative domains
«use»
«datastore»
Internet/Web pages
«use»
Recognition of the innovative logo for each domain
«save»
«datastore»
The innovative logo of companies
Receive the first page for each domain and extract the main text (company description)
«save»
«datastore»
The description of companies
Receive the first page for each domain and extract labels of links
«save»
«datastore»
The links labels of companies
Receive the three best innvotive documents from the domain and merg them to one big document
«save»
«datastore»
The big document of companies
Get for each company the company logo, description and big document
The company logo, description and big ducument
Merge the company logo and company description
Merge the company logo and company links label
The company big document
The company logo and company description
The company logo and comapny likns labels
The company big document
Classification by NB 1
Classification by NB 2
Classification by NB 3
Decision of the NB 1
Decision of the NB 2
Decision of the NB 3
Voting by a committee of classifiers
«save»
«datastore»
The decision of the classifiers committee

Fig. 3. The proposed classification system of innovative domains.

Next, the *Receive the first page for each domain and extract the main text (company description)* component processes the main text, i.e. tokenizes it to a simple feature set (a set of features of a type s_2) and saves it in *The description of companies* data-store $D_{description}$ using the binary representation. The *Receive the first page for each domain and extract labels of links* component gets all labels of links from the first page, i.e. it extracts the phrases between $< a >$ *(.+?)* $< /a >$ HTML tags and joins the extracted phrase to the single feature. We can extract the feature $f = our - company$ from $< a >$ *our company* $< /a >$ and transform it to $f^{`} = our - company$. The results are saved in *The links labels of companies* data-store $D_{labels\, of\, links}$ using the binary representation. The next component named *Receive the three best innovative documents from the domain and merge them to one big document* utilizes the Okapi BM25 to search the best innovative documents in the domain. This process is divided into two phases. In the first phase, we find three best documents, which are on the top of a rank $scoring(q, d)$, where $q = \{NP, NER\,labels, the\,named\,innovative\,phrases\}$ and $d \in D_{all}$. The NP is the *noun phrase*, NER are the recognized name entities (in a document we marked, using the Named Entity Recognition NER labeller, all the recognized entities, for example, the *Albert Einstein* was labelled as PERSON, etc.) and *the named innovative phrases* are manually marked "innovative" phrases like a *b-r*, *patent*, *venture-capital*, etc. We can interpret the query q as follows: find the documents that are saturated by the *noun*, NER, and *manually* created phrases. In the second phase, we merge three best documents from the domain into one big document. In this case we transform all NP, which occur in the document, into real phrases, e.g. *production-line*, *innovative-solution*, etc. The merged document is saved in *The big document of companies* data-store $D_{big\,document}$ using the TF representation. After creating all necessary data-sets and NB classification models (based on the data-sets mentioned above), we use the *Get for each company the company logo, description and big document* component to get the required company data to the classification process. Finally, in three parallel process, the appropriate data is merged and classified using the NB_1 (Bernoulli distribution model), NB_2 (Bernoulli distribution model), and NB_3 (Multinomial distribution model) respectively. The *Voting by a committee of classifiers* component creates the final decision by voting and saves the results into *data stores.* In our case, the most frequent label wins.

4 Innovative Domains Classification System - Evaluation

Data set. It is difficult to provide a relevant data set of labelled domains because the manual labeling is a very demanding and time-consuming task. Each domain must be assessed whether it is *innovative* or *no innovative.* Despite these problems, we decided to create an original test set due to the lack of such data in the other works. We labelled 2,747 domains and created three sets as follows:

- $D_1[2,747 \times 140,699]$ - each example contains a company description $D_{description}$ and its logo D_{logo}, which gives 140,699 features;
- $D_2[2,747 \times 140,271]$ - each example contains link labels originating from the first page of a company domain $D_{labels\,links}$ and its logo D_{logo}, which gives 140,271 features;
- $D_3[2,747 \times 663,015]$ - each example contains a big document created form three the most relevant documents selected form a company domain $D_{big\,document}$, which gives 663,015 features.

The classification quality is evaluated according to the 10-fold cross-validation procedure and measured by precision, recall, and F measure (in fact, it is F1 measure, which assumes an equal balance of precision and recall) [8].

Naïve Bayes classifiers and feature selection methods. The first set of experiments covered comparisons of various feature selection methods (see Subsection 2.2) by using Naïve Bayes classifiers based on the Bernoulli distribution model (NB_1, NB_2) and the Multinomial distribution model (NB_3).

Figure 4 contains the F-score values achieved by above classifiers. Namely, the results depicted in Fig. 4A are produced by the NB_1 on the set D_1, Fig. 4B shows the outcomes of the NB_2 on the set D_2, and Fig. 4C includes the results of the NB_3 on the set D_3. Both NB_1 and NB_2 classifiers use the Bernoulli distribution model, whereas the NB_3 uses the Multinomial distribution model.

The set D_1, which contains descriptions and logos of companies, gives better results in comparison to the set D_2 regardless of almost all methods used for feature selection. The set NB_3 causes higher differences in performance among methods of feature selection in comparison to two previous sets. The NB_1 classifier works the best in the case of using the χ^2 feature selection method (Fig. 4A). On the contrary, the classifier NB_2 produces the best results when using $Fisher$ feature selection method. However, the differences among tested methods are rather not very significant in this case (Fig. 4B). Finally, the classifier NB_3 performs the best when using the $Fisher$ feature selection method (Fig. 4C).

The diversified classification committee. The second series of experiments was intended to compare the best Naïve Bayes classifiers from the previous experiments with the proposed classification committee.

Figure 5 compares the F measure values of the committee and three the best NB classifiers according to the previous experiments. The committee outperforms every single classifier when the number of features varies between 600 and 4,000. The further increase in the number of features leads to the decrease of F measure of the committee. Moreover, there are observed higher F measure values of the NB_1 classifier with the χ^2 feature selection method than the committee. However, the analysis of precision and recall (Fig. 6) shows that the increase in the number of features causes overfitting of almost all classifiers and the committee. This is observed as the precision increase and the recall decrease.

The precision increase and the recall decrease are equivalent to the increase of correct decisions that the domains really represent innovative or no innovative companies and concurrently the increase of the number of really innovative domains that are not marked as innovative companies. This is improper behavior because many innovative companies are not selected, whereas it is better for our purposes when some non-innovative companies are treated as innovative (lower precision) but the high number of truly innovative businesses is found on the Internet (higher recall).

Worth to note are the results of the NB_3 with the set D_3 created by the proposed information retrieval system. This classifier outperforms the committee when the number of features is very high (Fig. 5) and, at the same time, it is robust to overfitting (Fig. 6). However, the classifier requires two times higher number of features than the committee.

Although each classifier in the committee uses the different feature space, the number of features is the same for all of them. It is the sub-optimal solution because we can easily notice that the classifiers produce the best results in the distinct ranges of feature numbers (Fig. 6). The optimal system may involve an adaptive selection of the features number for each classifier in the committee. However, our approach is simple and produces the acceptable results.

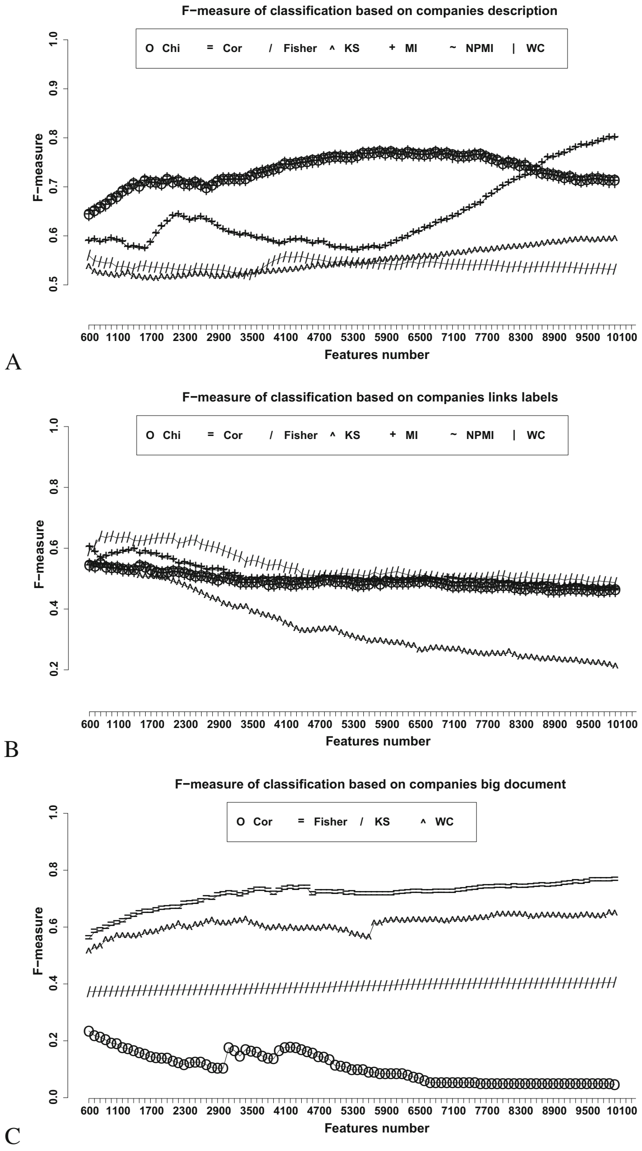

Fig. 4. F-measure values achieved by (A) the Naïve Bayes classifier NB_1 on the set D_1, (B) the Naïve Bayes classifier NB_2 on the set D_2, and (C) the Naïve Bayes classifier NB_3 on the set D_3 when using various methods of feature selection.

We can conclude that the proposed committee produces the most stable decisions, even if some its classifiers are unstable is some range of features, e.g. NB_2 (see Fig. 5). Moreover, the proposed information retrieval system improves

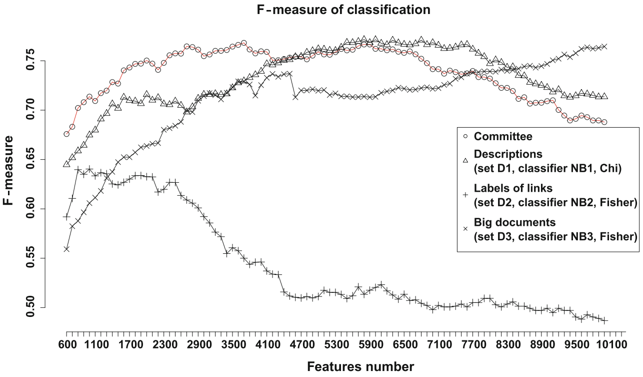

Fig. 5. F measure of the best Naïve Bayes classifiers and the committee.

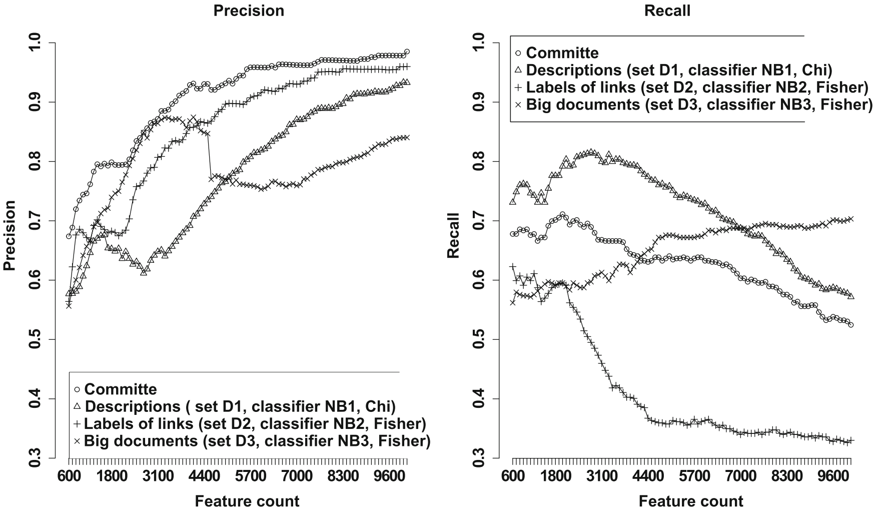

Fig. 6. Precision and recall of the best Naïve Bayes classifiers and the committee.

the classification quality, and, what is more important, it prevents both classifiers and the whole committee from overfitting.

5 Conclusions

The objective of this work was to propose a classification method of potentially innovative domains acquired from the Internet to estimate whether they represent companies that are innovative or not. Since it is not clear which attributes indicate innovativeness of a company, the classification model had to utilize the numerous features without exact information regarding their importance. The preliminary tests showed that a single classifier is not suitable for this task.

Thus, we have designed and tested a diversified classification committee that is composed of Naïve Bayes classifiers based on Bernoulli and Multinomial feature distribution models.

During the preliminary experiments we found out that the use of a whole domain as input to the classification system is inappropriate. Thus, we have applied retrieval methods at the preprocessing phase to extract the most innovative documents from analyzed domains. These methods were the Okapi BM25 ranking function and the Named Entity Recognition labeler. The proposed information retrieval system improved the classification quality. Moreover, it prevented both the classifiers and the whole committee from overfitting.

In the experiments, the performance of simple Naïve Bayes classifiers and the proposed system was analyzed in respect of feature numbers. It has to be noted that the number of features was selected according to Fisher and χ^2 distributions. We can conclude that the proposed committee produces stable decisions, even if some its classifiers are unstable is some range of features. The classification quality achieved by the system is acceptable for our purposes. However, it may be possible to improve the performance by iterative training of classifiers using manually prepared training data. Moreover, the further research may involve an adaptive selection of the features number for each classifier in the committee.

We believe that the proposed classification system is suitable for such classification tasks were classification models have to deal with unstructured data. Thus, this approach may be applied to other classification tasks.

Acknowledgements. We would like to thank the anonymous reviewers for their comments that significantly helped to improve the manuscript. Moreover, we are grateful for Krzysztof Wolinski and his team support in labeling of data.

References

1. Morphological analyser morfeusz. http://sgjp.pl/morfeusz/morfeusz.html.en. Accessed 28 Oct 2015
2. Bellotti, T., Nouretdinov, I., Yang, M., Gammerman, A.: Feature selection, pp. 115–130. Elsevier (2014)
3. Bouma, G.: Normalized (pointwise) mutual information in collocation extraction. In: Biennial GSCL Conference 2009, Tübingen, pp. 31–40 (2009)
4. Figueiredo, F., Rocha, L., Couto, T., Salles, T., Gonçalves, M., Meira, W.: Word co-occurrence features for text classification. Inf. Syst. **36**(5), 843–858 (2011)
5. Hand, D., Smyth, P., Mannila, H.: Principles of Data Mining. MIT Press, Cambridge (2001)
6. Li, S., Xia, R., Zong, C., Huang, C.: A framework of feature selection methods for text categorization. AFNLP **2**, 692–700 (2009)
7. Liu, B.: Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data. Data-Centric Systems and Applications. Springer, Heidelberg (2006)
8. Manning, C., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)

9. Nakatsuji, M.: Identifying novel topics based on user interests. In: Elçi, A., Koné, M.T., Orgun, M.A. (eds.) Semantic Agent Systems. SCI, vol. 344, pp. 273–292. Springer, Heidelberg (2011)
10. Nakatsuji, M., Miyoshi, Y., Otsuka, Y.: Innovation detection based on user-interest ontology of blog community. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 515–528. Springer, Heidelberg (2006)
11. Nakatsuji, M., Yoshida, M., Ishida, T.: Detecting innovative topics based on user-interest ontology. Web Semant. Sci. Serv. Agents World Wide Web **7**(2), 107–120 (2009)
12. Piasecki, M., Szpakowicz, S., Broda, B.: Toward plWordNet 2.0. Principles, Construction and Application of Multilingual Wordnets, pp. 263–270 (2010)
13. Przepiórkowski, A., Buczyński, A.: Shallow parsing and disambiguation engine. In: Proceedings of the 3rd Language and Technology Conference, Poznań (2007)
14. Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M., Gatford, M.: Okapi at TREC-3. In: Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, pp. 109–126 (1994)
15. Romanov, D., Ponfilenok, M., Kazantsev, N.: Potential innovations (new ideas/trends) detection in information network. Int. J. Future Comput. Commun. **2**(1), 63–66 (2013)
16. Sammut, C., Webb, G.: Encyclopedia of Machine Learning. Springer, New York (2011)
17. Schenker, A., Bunke, H., Last, M., Kandel, A.: Graph-Theoretic Techniques for Web Content Mining. World Scientific Publishing, Singapore (2005)
18. Schurmann, J.: Pattern Classification - A Unified View of Statistical and Neural Approaches. Wiley, New York (1996)
19. Solka, J.: Text data mining: theory and methods. Statist. Surv. **2**, 94–112 (2008)
20. Woliński, M.: Morphological tagset in the ipi pan corpus, Polonika, pp. 39–54 (2004)
21. Wolinski, M.: Morfeusz - a practical tool for the morphological analysis of polish. Intell. Inf. Process. Web Min. Adv. Soft Comput. **35**, 511–520 (2006)