Jurnal
Teknologi

Full Paper

A CATEGORY CLASSIFICATION ALGORITHM FOR INDONESIAN AND MALAY NEWS DOCUMENTS

Jafreezal Jaafar*, Zul Indra, Nurshuhaini Zamin

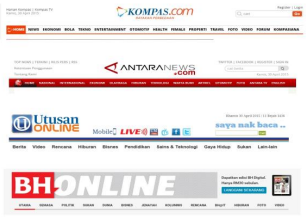
Deparment of Computer and Information Sciences Universiti Teknologi PETRONAS, 32610 Bandar Seri Iskandar, Perak, Malaysia

Article history

Received
30 November 2015
Received in revised form
27April 2016
Accepted
23 February 2016

*Corresponding author
jafreez@petronas.com.my

Graphical abstract



Abstract

Text classification (TC) provides a better way to organize information since it allows better understanding and interpretation of the content. It deals with the assignment of labels into a group of similar textual document. However, TC research for Asian language documents is relatively limited compared to English documents and even lesser particularly for news articles. Apart from that, TC research to classify textual documents in similar morphology such Indonesian and Malay is still scarce. Hence, the aim of this study is to develop an integrated generic TC algorithm which is able to identify the language and then classify the category for identified news documents. Furthermore, top-n feature selection method is utilized to improve TC performance and to overcome the online news corpora classification challenges: rapid data growth of online news documents, and the high computational time. Experiments were conducted using 280 Indonesian and 280 Malay online news documents from the year 2014 – 2015. The classification method is proven to produce a good result with accuracy rate of up to 95.63% for language identification, and 97.5%% for category classification. While the category classifier works optimally on n = 60%, with an average of 35 seconds computational time. This highlights that the integrated generic TC has advantage over manual classification, and is suitable for Indonesian and Malay news classification.

Keywords: Text classification, Text Mining, Information Retrieval

Abstrak

Klasifikasi teks (KT) adalah kaedah yang lebih baik untuk menguruskan maklumat kerana ia memudahkan pemahaman dan interprestasi isi kandungan. Ia melibatkan peletakan penanda kepada kumupulan dokumen teks yang sama. Walaubagaimanapun, bagian KT untuk dokumen-dokumen berbahasa Asia adalah lebih sedikit berbanding dengan dokumen-dokumen berbahasa Inggris, malah sangat sedikit jumlahnya untuk kajian artikel berita. Selain daripada itu, kajian KT untuk mengklasifikasikan dokumen teks yang mempunyai morphology yang sama seperti bahasa Indonesia dan bahasa Melayu adalah masih berkurangan. Oleh itu, matlamat kajian ini adalah untuk membina algoritma gabung KT yang berupaya untuk mengenal pasti jenis Bahasa dan seterusnya berupaya untuk mengklasifikasikan kategori untuk dokumen berita tersebut. Disamping itu, kaedah pemilihan ciri top-n telah digunakan meningkatkan prestasi KT dan untuk mengatasi cabaran pengklasifikasian korpus berita atas talian: pertumbuhan pesat data dokumen berita dalam talian, dan masa pengkomputeran yang tinggi. Eksperimen telah dijalankan dengan menggunakan 160 bahasa Indonesia dan 160 bahasa Melayu dokumen berita atas talian dari tahun 2014 hingga 2015. Kaedah integrasi ini telah dibuktikan untuk dapat menghasilkan keputusan yang baik dengan ketepatan sehingga 95.63% untuk pengenalpastian bahasa dan 97.5% untuk pengklasifikasian kategori teks. Bagi

pengkomputeran masa, pengumpul berita automatik (PBA) memerlukan masa kurang daripada 5.52 saat untuk mengumpukan data, sedangkan algoritma pengenalpastian bahasa memerlukan masa kurang daripada 5.52 saat untuk mengenalpasti sebuah bahasa. Manakala bagi pengklasifikasi kategori pula, keputusan yang baik dicatatkan pada n=60% dengan purata 35 sec masa pengkomputeran. Perbandingan kami dengan semakan manusia, telah mendapati bahawa algoritma pengenalpastian bahasa berjaya
Kata kunci: Klasifikasi Teks, Perlombongan Teks, Capaian Maklumat.

© 2016 Penerbit UTM Press. All rights reserved

1.0 INTRODUCTION

At present, it can be said that Internet has been becoming a major source of information and knowledge for our life. Since it was introduced to the public, the amount information in Internet has been increased very quickly. In 1994, the World Wide Worm (one of the earliest web search engine) claimed that it had indexed 110,000 web documents [1]. By the last of 1997, total of 2 million to 100 million had been indexed by web crawler which was known as the top search engine at that time [1]. Furthermore on March 2004 based on Google claim, there are 3 billion web documents had indexed by Google at the end of 2001, 4.28 billion on March 2004 [2]. The number of web pages in Internet will continue to increase reported in 2010. [3] revealed that 7 million new web pages being added daily.

Due to the rapid increased of information and knowledge on the Internet, the online news websites on Internet is also increasing very quickly. Hence task of extracting the related information on the online news website is a challenge. The digital news readers need a tool to ease them to find information which is relevant to their interest. One of techniques to ease navigation is to classify news using automatic text classification algorithm.

This research contributes an improved text classification (TC) algorithm to allow Internet user to more easily find information around the online news. In some literature review TC is classified as a technique under the Text Mining area [4-6]. This means that TC is considered similar to information retrieval. However, a literature described that TC is not similar to information retrieval but TC is only a sub discipline of information retrieval [7]. Other researcher [8], claimed that the task to perform TC requires multi-technique i.e. IR, Machine Learning, and Natural Language Processing at the same time. The main goal of TC is to assign the documents into one or more categories. The implementation of TC may be in the form of text dialogues, scientific writings, or any textual data existing online [9]. As one huge information source, web pages are among the most heavily exploited source for data collections which include the online news document. Most online news documents are freely accessible. However on the online publication, the classification of online news documents becoming more challenging when the properties of data attributes such as length [10] and writing style are varies.

To achieve the objectives of the research, Indonesian and Malay news documents are chosen as case study since research about text classification for documents in these language is relatively less compared to English [11]. Text classification for Indonesian and Malay news documents should become an important concern due to the huge number of speakers which is spread in several countries such as Indonesia, Malaysia, Brunei, Thailand and Singapore. As the official language of the republic of Indonesia, the users for Indonesian language are estimated to be 220 million speakers. Indonesian language is closely related to the Malay language in Malaysia, Singapore, Brunei since these languages are derived from the literary of the Malay dialect [12]. Thus, the users for these languages become more numerous if the Malay spoken users are counted. Moreover a research in [13], found that Indonesian/Malay language had achieved ninth rank which population; variability; distribution; religious circumstances and linguistic aspects were the factors considered. Thus, the main objective of this research is to develop a generic TC algorithm for Indonesian and Malay news document.

In the different way with the existing TC algorithm which is only used to classify document for one language, this generic TC algorithm is intended to be able classify Indonesian and Malay news document. As Indonesian and Malay language is very similar, it supposed to be that TC algorithm can classify the document for both of these languages. A new language classification algorithm is developed prior to TC process. This language classification algorithm is assigned the news document into the correct language whether the news document belongs to Indonesian or Malay language. Thus, the final output of this research is a generic TC algorithm which is able to classify language and category for Indonesian and Malay news document.

2.0 RELATED WORKS

The task of classification is commonly associated with Machine Learning –the development of systems

which enables computer to learn from a given set of rules and samples to improve its performance through each new. Since the range of algorithms for text classification is vast, the literature studied in this thesis mostly focus on the TC algorithm which is widely used on classifying Indonesian and Malay document. A review of the related works of TC process for Indonesian and Malay document is explained in the next subsections.

1.1 Classification for Indonesian and Malay Documents

One of the earliest for TC in 2009 was conducted by Asy'arie and Pribadi [14]. Asy'arie and Pribadi conducted a study on news classifier using the Naive Bayes classification method. For each document, terms are pre-processed, stemmed, and weighted. Classification is done by calculating the probability between corpus collection and the label. The label assigned to the document is categorised with maximum posteriori. The classifier managed to reach more than 90% F-measure average with a recall rate up to 93.75%. The weak point in this effort is that the method hardly processes news with more than 1,000 words and simply produces null as its value. Since news documents can be two to six paragraphs long, thus it is difficult to classify the news documents using this algorithm.

One of the most commonly used techniques in text classification is SVM. Liliana, et al. [15] utilised this algorithm on Indonesian news collections and achieved an average accuracy of 85%. The SVM classifier uses an Optimum Separating Hyperplane, which is designed to separate objects into their fitting class. If an object is located on the separator line, it becomes a support vector. Sharing similar issues with [16], the parameter gamma SVM this method is defined by the user without specific criteria.

One approach using the *k*-NN method was developed in 2012 by Firdan et al. [16]. Even though the end product is emotion classification, the methodology bears a similar overall structure with the algorithm described in this thesis. It first pre-processes the training data, stems and then weights the data collection. Once the data is weighed, the *k*-NN phase begins. This algorithm classifies an object by first selecting k other nearest objects ("neighbours") surrounding it, then assigning the object to the most frequent neighbour. The fundamental problem with *k*-NN is its dependency on the majority of members of the class, hence making it unsuitable for collection with highly diverse data. Another issue is the difficulty in determining where the exact number for k should start, given that there has never been a specific rule except for those determined experimentally. As seen in [16], k used in the tests may vary from two to 60, while the optimum k yielded is 40.

An improvement of the above method was soon introduced by [17]. A combination with k-means clustering algorithm was aimed at reducing high calculation complexity and helping the classifier to be less dependent on the size of the training data. This is done by clustering the training documents into k number of clusters, while each cluster is represented by a chosen centroid. Thus, instead of calculating similarity between the complete set of training data with the sample, classification would only deal with respective centroids, producing more efficient computation. The shortcoming of this method lies in the nature of most clustering methods, as the best starting number for cluster has always been vague. The algorithm yields 85% F-measure.

Another implementation of a Bayesian classifier was recently studied in early 2013 [18]. The unique approach on Naive Bayes method in this paper was its usage to classify personality in a social website by studying the personality paragraph written by the user in the page. Of course, with user-defined materials it is even more difficult to uniform the collection due to the style of writing, which may differ significantly among users. Although the method satisfied a 92.5% accuracy, it relies excessively on the number of training documents it studies [18].

The number of text classifier even lesser for Malay text document. One of few studies for Malay text classifiers was conducted by Noah and Ismail [19]. Noah and Ismail proposed an automatic classification of Malay proverbs using two models of Naive Bayes algorithm i.e. multinomial and multivariate model. To conduct the experiment, the dataset for this study was divided into dataset with stop words and dataset without stop words. Later on, the experiment was conducted by using three cases of Malay proverbs i.e. proverb alone, proverb with the meaning and proverb with the meaning plus the example sentence. However, this works achieved 72.2% and 68.2% maximum accuracy for both of the models respectively with no stop words using proverb with the meaning and example sentence which are not good enough.

Another study for Malay text classifier was done by Alshalabi, et al. [20]. This study compared the result of Malay text classifier by using *K*-NN, Naive Bayes and N-gram algorithm. These TC algorithms are combined with two methods of feature selection i.e. IG and Chi-square in order to test the efficiency. Using the Macro-averaged (Macro-F1), the experimental result for this study showed that the *K*-NN with the Chi-square achieved the best performance (Macro-F1 = 96.14). The summary of TC for Indonesian and Malay is shown in Table 2.13.

From this literature review, it is found that there are three supervised TC algorithm which is widely used on classifying Indonesian and Malay document. The further description for these three TC algorithms is discussed in the next subsections.

1.2 Naïve Bayes

Naïve Bayes (NB) is one of the most popular algorithms for text classification which derived from Bayes theorem. The principle of Bayes theorem is the assumption that all the properties of a given class are independent in such a way that the existence of a feature does not affect other attribute. NB classifier is also known as a generative text classification since it generates a probabilistic model [21]. The generative model is then fitted by employing a set of labelled training data [22]. The basic notion of NB classifier is the calculation of the posterior probability between the documents with a given set of classes [23]. The class with the highest posterior probability is then selected to be class for the documents.

Due to its simplicity and ease of use, Naïve Bayes remains popular classification algorithm for text classification among researchers [24]. The efficiency of Naïve Bayes is due to this algorithm only requires once scanning process to scan the entire training data during data training [25]. Nonetheless, this algorithm has issue in sensitivity towards the number of training documents [26]. The performance of Naïve Bayes is reduced when the amount of training data in each category is imbalanced. Similarly, Naïve Bayes does not work well on small data set [27] and does not perform as good on rare categories [22, 28]. Moreover, it was revealed that Naïve Bayes can't work well with less than 10 training data in one class [29], despite efforts on improving or combining it with other methods [30-32]. Thus, it can be concluded that the performance of Naïve Bayer is highly depend on training data set. Naïve Bayer requires a large training data set since it does not perform well for small training data and category.

1.3 Support Vector Machine

Support Vector Machine (SVM) is a relatively new algorithm for TC. SVM was introduced by Cortes & Vapnik in 1995 and first used in TC by Joachim in 1998 [33]. SVM is binary classifier which is based on a vector space method. Generally, the main concept of SVM is based on the *Structural Risk Minimization*, which is based on finding the lowest true error through a hypothesis. The algorithm works by learning from positive and negative examples. Although the use of positive and negative examples is relatively rare in the method of text classification, the method has been utilized many times in the text classification field [34-37]. Apart from that, since SVM is binary classifier, it implies a potential issue when presents a multi-label classification task, the task associated with news classification.

SVM is known as reliable TC algorithm which produces good results [38]. Moreover, SVM is a successful method which can simultaneously minimizes the empirical classification error and maximizes the geometric margin. Nevertheless, since it is impossible to resort the domain of news classification to only two classes, so the utilization of SVM is not preferable even when it is already capable of handling multi-class classification. The employment of SVM for multi-class problem can be done by creating an *n* number of SVM classifiers [38], with *n* being the number of the classes used as classes in the classification. Even so, comparison between an *n* SVM with a single multi-class classifier such as Naive Bayes or K-NN remains unjust [39]. This is because in a multi-class classification issue, positive examples are those within the category they belong to, while the rest of the collection which belong to the rest of the category set are considered to be negative examples. This means that the number of negative examples would be significantly larger that of positive examples. [40] has suggested that the performance can be adversely affected by such skewness.

1.4 K-Nearest Neighbours

The *k*-nearest neighbour (*k*-NN) classifier is one of the earliest algorithm in the development of TC [41]. Since it was first introduced, *k*-NN has become one most popular algorithm in TC due to its low implementation cost and high degree of classification effectiveness [25, 42]. Over the decades, the *k*-NN has been widely used in various types of classification [43, 44]. *k*-NN is also known as type of on-demand classifier which does not build a classification model (Baeza-Yates & Ribeiro-Neto, 1999). Contrary to the other TC algorithm, the classification task in type of on-demand classifier is performed only at the moment a new document d_j is given to the classifier.

However, *k*-NN has problem in time to perform classification when the numbers of training data are of very great size [45]. *k*-NN requires a lot of time to calculate the similarity between the training data and test data. On the contrary, when the number of training data is not large enough, the performance *k*-NN algorithm is no longer optimal. To deal with these problems, the feature selection can be implemented to reduce the feature space or vector which represents the document. The feature selection will be reduced the time to perform TC by selecting a subset of all feature to represent the documents (Kira & Rendell, 1992). Another problem in *k*-NN algorithm is in determining the best of parameter K. A study by Geng, et al. [46] reveal that the performance of *k*-NN algorithm is depend on the value of parameter *k*. The appropriate value parameter *k* is then required to achieve the best performance of TC.

According to [46], the *k* -NN achieves moderate performance for small value of K due to fact that the smaller value of *k*, the lesser information

will be obtained from the training. On the contrary, if the value of *k* is increased, the performance of the *k* -NN is increased too. However, the performance of the *k* -NN will decrease if the value of *k* exceeds a certain threshold. This phenomenon was revealed by Callut, et al. [47], too many neighbours which have been used lead to the occurrence of noises to the information obtained from the training data [47].

2.0 METHODOLOGY

The main objective of this research work is to develop a generic TC algorithm for Indonesian and Malay news document. *k*-NN algorithm is utilise to classify the news document into appropriate category. The basic concept of the classification algorithm is the calculation of similarity degree between documents to be classified with the pre-defined category [45]. The predefined category is described by the keywords from supporting documents such the training document.

The *k*-NN algorithm classifies category for the test news document by first selecting *k* other nearest news document ("neighbours") surrounding it and then assign the test news document to the most frequent neighbour [48]. First, it pre-processes the test news document to generate the weight for each term in the test news document. Once the test news document is weighted, the *k* -NN starts to classify category for the test news document.

Suppose *j* denotes the number of training categories *d* (d_1, d_2, d_3 ... d_j) and *N* is the total number of documents in the training samples. Shortly, the steps in classification using *k* -NN can be described as follows [49]:

1. Define value of *k*
2. Transform the test document, say *X*, into the same vector space as the training documents and generate the weight for the test document
3. Select the keywords for the test data.
4. Calculate the similarities between *X* and training document.
5. Sort the similarity degree and choose *k* documents with the largest similarities from *N* number of similarities. These documents are then considered as the collection for *X*.
6. Choose the category from *k* documents with the majority member document to be category for the test document.
7. Suppose there are more than one member of *k* which share a category, then accumulative the similarity degree for each category.
8. Choose the category with the biggest accumulation similarity degree to be category for test document

The flowchart for *k*-NN algorithm is shown in Figure 1.

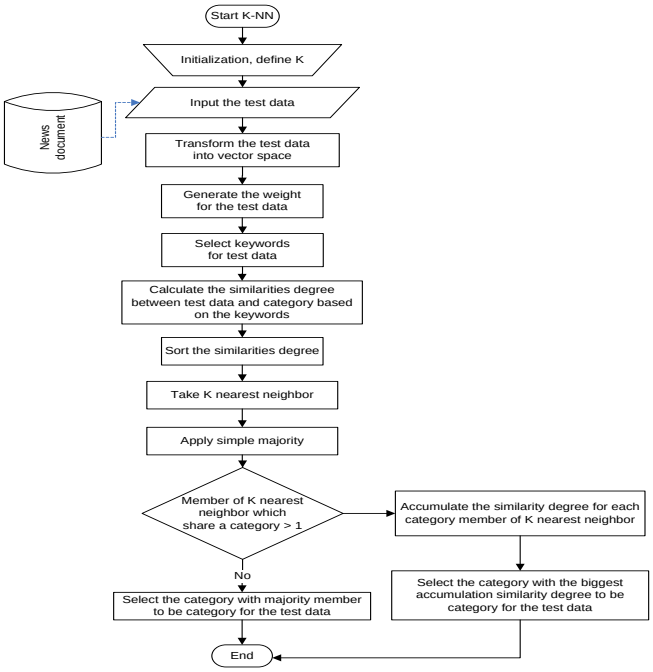


Figure 1 k-NN Algorithm Flow Chart

The similarity degree between the test data and the training data is calculated based on their keywords. Thus, prior to the similarity calculation, the keywords for the test data are selected using same method described in the training phase. Once the keywords are retrieved, the similarity calculation takes places.

In this method, the similarity calculation is done by using cosine similarity measure. The Cosine Similarity measure is the most popular tool to calculate document similarity based on the Vector Space Model (VSM). The weighting process in pre-processing phase generates vector for each document from sets of terms with associated weights. The generated vectors would be chosen as keywords using top-*n* algorithm and then the keywords are used in the cosine similarity formula for similarity scoring.

$$sim_{cosine}(a,b) = \frac{\sum_{t \in a \cap b}(w_{a,t} \times w_{b,t})}{\sqrt{\sum_{t \in a} w_{a,t}^2} \times \sqrt{\sum_{t \in b} w_{b,t}^2}}$$

Where $a \cap b$ gets the common keywords between document a and document b. $t \in a\ (or\ b)$ means t is a unique term in document $a\ (or\ b)$. $w_{a,t}$ and $w_{b,t}$ are the weights of term t in document a and b that have been computed in weighting process using TF-IDF algorithm.

For example, a test news document has content about the competition between Google and Facebook in mobile advertisement. In this case,

the classifier selects the following 10 keywords: "google", "*pasar*" (market), "browsing", "online", "*iklan*" (ad), "network", "user", "facebook", "mobile", "*sosial*" (social). "google" is denoted as k_1, "*pasar*" as k_2, and "*browsing*" as k_3 respectively until "*sosial*" as k_{10}.

Table 1 TF-IDF for keywords

	Document Keyword	**tf**						**idf**	**wdt=tf*idf**				
		d	**d_1**	**d_2**	**d_3**	**d_4**	**df**	**log(n/df)**	**d**	**d_1**	**d_2**	**d_3**	**d_4**
k_1	google	3	0	0	2	0	2	2.7	8.1	0	0	5.4	0
k_2	*pasar* (market)	1	2	1	1	0	4	2.4	2.4	4.8	2.4	2.4	0
k_3	browsing	2	0	0	3	0	2	2.7	5.4	0	0	8.1	0
k_4	online	1	0	0	4	0	2	2.7	2.7	0	0	10.8	0
k_5	*iklan* (ad)	3	1	0	0	0	2	2.7	8.1	2.7	0	0	0
k_6	network	1	0	0	1	2	3	2.52	2.52	0	0	2.52	5.04
k_7	user	1	0	0	0	0	1	2.7	2.7	0	0	0	0
k_8	facebook	2	0	0	1	0	2	2.7	5.4	0	0	2.7	0
k_9	mobile	4	0	0	3	0	2	2.7	10.8	0	0	8.1	0
k_{10}	sosial	1	1	0	0	1	3	3	3	3	0	0	3
	Total number document (n) = 1000												

Table **1** show a collection data which consist of 1000 document. Test document is denoted as *d* and training document are denoted as d_i. These training document are classified into two classes, C_1 (technology) and C_2 (economy), where d_1, d_3 are member of C_1 and d_2, d_4 are member of C_2. The *tf* is the frequency of the keyword in each document and the *df* is document frequency which is defined to be the number of documents in the data collection that contain the keywords. The weighted for the keyword is denoted as *wdt*. The weighted value is result of the multiplication between frequency of the keyword and inverse document frequency (*idf*). Once the weighted for the keywords has been calculated, the similarity between these keywords for each document is ready to be calculated by using Cosine Similarity measure.

Prior to the calculation of similarity degree, the multiplication results between the weighted of keywords in test data and training document need to be calculated and then these results are cumulated. Furthermore, the length vector for the document is also need to be calculated by squaring the weighted of keywords, cumulate it, and then calculate the root square to normalize it. The illustration of these processes is shown in Table **2**.

Table 2 Length Vector for Keywords

wdt*wdi				**The length vector**				
d_1	**d_2**	**d_3**	**d_4**	**d**	**d_1**	**d_2**	**d_3**	**d_4**
0	0	43.74	0	65.61	0	0	29.16	0
11.52	5.76	5.76	0	5.76	23.04	5.76	5.76	0
0	0	43.74	0	29.16	0	0	65.61	0
0	0	29.16	0	7.29	0	0	116.64	0
21.87	0	0	0	65.61	7.29	0	0	0
0	0	6.3504	12.7008	6.3504	0	0	6.3504	25.4016
0	0	0	0	7.29	0	0	0	0
0	0	14.58	0	29.16	0	0	7.29	0
0	0	87.48	0	116.64	0	0	65.61	0
9	0	0	9	9	9	0	0	9
42.39	5.76	230.8104	21.7008	341.8704	39.33	5.76	296.4204	34.4016
root square				18.48974	6.271363	2.4	17.21686	5.865288

Table **2**, the multiplication results between the weighted of keywords in the test data and the training document is denoted as *wdt.wdi*. The last step in similarity degree calculation is calculating the similarity between keywords in test data and categories. The similarity calculation is done by dividing the multiplication results between the weighted of keywords to the multiplication of normalized length vector for the documents.

For example, the similarity calculation between test data and d_1, where the value for the multiplication results between the weighted of keywords are 42.39 and the multiplication of normalized length vector for test data and c_1 is 115.956, is shown in formula below:

$$Similarity\ (d, c_1) = 42.39 / (18.48974*6.271363) = \mathbf{0.366}$$

Table 3 Similarity calculation between test data and training document

	d_1	d_2	d_3	d_4
d	0.366	0.13	0.725	0.2

The overall similarity degree between test data and all categories is shown in
Table 3. The next step is to sort the similarity degree in ascending order as shown in
Table **4**.

Table 4 Ordering similarity calculation

	d_3	d_1	d_4	d_2
d	0.725	0.366	0.2	0.13

Once all the similarity degrees are sorted, the next step is choosing the *k* documents with the highest similarity from *N* number of similarities. For example in this case, the number of *k* is 3 (k=3). Thus the test data is classified into C_1 since C_1 are represented by two documents (d_1 and d_2) for C_1 and C_2 is represented only by one document (d_3).

3.0 SYSTEM PERFORMANCE EVALUATION

The objective of this testing is to investigate the *k*-NN category classification algorithm and top-*n* feature selection method performance for Indonesian and Malay news document. Moreover, to analyse how the top-n method improves the performance of the category classification algorithm.

As the starting point, the testing document is input to the software. Before classification, the *n* value for top-n method and *k* value are set to get the best performance of category classification algorithm. Finally, the performance of category classification algorithm with several conditions of *n* value and *k* number is measured and analysed.

3.1 Dataset for Category Classification

The dataset for this experiment consist of two type data, namely training data and testing data.The training data is stored as supporting documents for the predetermined categories. Through observations, only four categories always appear for all news website. These categories are the source of classification during the training stage, as well as the ground truth for the first set of experiment. While the testing data is the news documents which is used to examined the performance of category classification algorithm.

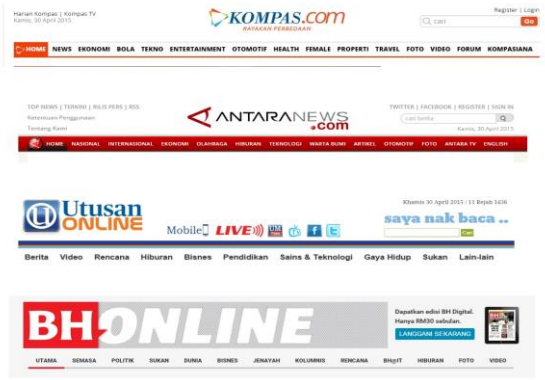


Figure 2 The Categories List off News Website

As in Figure 2, only four categories always appear for all news website. These categories are ekonomi or bisnes (*economy*), olahraga or sukan (*sport*), hiburan (*entertainment*) and teknologi or sains&Teknologi or BH@IT (*technology*). These categories have different spelling in each news website. The economy category in Indonesian news website (KOMPAS and ANTARA) is referred to ekonomi, while in Malay news website it is referred to bisnes. In Indonesian news website, sport category is referred to olahraga, while in Malay news website it is referred to sukan. The entertainment category is referred to entertainment in KOMPAS, but in other news website it is referred as hiburan. Moreover, the technology category has different spelling for each news website. In Indonesian news website, this category is referred to tekno in KOMPAS, but it is referred as teknologi in ANTARA. In Malay news website, it is referred sains&Teknologi in UTUSAN, while it is referred as BH@IT in BERITA HARIAN. Although these categories have different spelling, these categories are same in meaning.

To start with the training data, 25 documents are used for each category. Thus, 200 documents are used as dataset for training data. However, instead

of downloading the news on date or time basis, the news is downloaded rather randomly. This is because that the classifier is hoped to be able to study a wide range of news characters, hence improving the sparse data issue. The list of training data is shown in Table **5**.

Table 5 Training Data for Category Classification

	Indonesian News Website		Malay News Website	
	KOMPAS	ANTARA	UTUSAN	BERITA HARIAN
economy	25	25	25	25
sport	25	25	25	25
entertainment	25	25	25	25
technology	25	25	25	25

In testing data, 160 news documents are used as dataset. These documents consist of 10 news document for each category for news website. In different way with training data collection which is downloaded manually form the news website, the ADC is utilised for testing data. The ACD algorithm is utilised to retrieved document automatically from URL of news webpage in order to ease data collection for testing data. By using the ADC, there are two attributes which are retrieved from the webpage. These attributes are the title and content of the news document. Thus the output of ADC which is used as testing data is the title and the content of the webpage. The list of testing data is shown in Table **6**.

Table 6 Testing Data for Category Classification

	Indonesian News Website		Malay News Website	
	KOMPAS	ANTARA	UTUSAN	BERITA HARIAN
economy	10	10	10	10
sport	10	10	10	10
entertainment	10	10	10	10
technology	10	10	10	10

At the same way with language identification experiment, the experiment for category classification also uses the words dictionary and the stop words. The words dictionary is used during stemming process to transform the word in the document to its root word. The result of stemming process is called as the term.

Table 7 Dataset for Category Classification

Training Document	160
Testing Document	80
Terms for training documents (words)	43,954
Stop Words (words)	1,442
Root Words Dictionary (words)	35,944

Table 7 shows the summary dataset for experiment of category classification. During the stemming process, each word in the document is checked into root words dictionary. If the word is found in the root words dictionary, it means that the word has reached its root form (lemma). When this stage is reached, the stemming process stops.

3.2 The result

The experiment of category classification in this research is conducted in two conditions. Firstly, the experiment is conducted by varying value of *k* for *k*-NN algorithm to get the best performance of category classification. As described in chapter 3, the best accuracy for *k*-NN algorithm is achieved by varying value of *k*. Once the best performance of category classification is achieved, the value of *k* with the best accuracy is recorded and then used for second condition experiment. Secondly, the experiment is conducted by implementing the top-n feature selection method to enhance the performance of category classification.

In the first condition of the category classification experiment, the value of *k* is varied from 3 to 11. These *k* values are chosen because they are only used to test whether the accuracy of classifier improves as the *k* value selected increases and to find the best value of *k*. Afterwards, the result of each classification using the value of *k* is compared .From a series of experiment, it is found that these numbers of selected *k* value improve the classifier most optimally until the value of *k* exceeds a certain value. In this research, the best performance of classifier is achieved at value of $k = 8$. The results of category classification for various values of *k* are explained in Figure **3**.

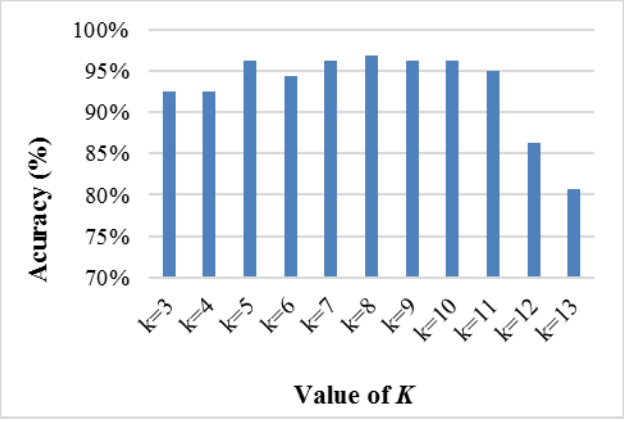


Figure 3 Result of Category Classification for Various k

The results reveal that the accuracy of classifier increased if the value of *k* is increased too. When the value of *k* is set at 3 which is the smallest *k* value in this research, the classifier achieved accuracy at 92.50%. Albeit it is a good accuracy, this accuracy is the lowest accuracy for the classifier. As seen in Figure 4.16, the classifier produced a better accuracy by increasing the value of *k*. The best accuracy of classifier is achieved at 96.88% by setting the *k* value at 8. Once the value of *k* is set bigger than 8, it is found that the performance of classifier became worse.

The possible reason for this trend is explained in the chapter 3. By selecting small value of *k*, the classifier would produce a bad performance due to fact that the smaller value of *k*, the lesser information will be obtained from the training. On the other hand, if the value of *k* is increased, the performance of the *k* -NN is increased too. However, the performance of the *k* -NN will decrease if the value of *k* exceeds a certain threshold.This phenomenon was revealed by Callut et al. (2008), too many neighbors which have been used lead to the occurrence of noises to the information obtained from the training data.

Based on the result of this experiment which is shown in Figure 4.16, it is found that the classifier can perform well to classify the category for the Indonesian and Malay news document. The classifier achieves a good accuracy level with average accuracy above 90%. Moreover, it can be concluded that the best value of *k* to be selected is 8 by achieving the best accuracy at 96.88%. This value of *k* is then used in the second condition of experiment by implementing the top-n feature selection method.

The next experiment of category classification is conducted by implementing the top-n feature selection method. Although the classifier in the first condition achieved a good accuracy, the classifier still has a problem in term of speed to classify the news documents. The classifier took almost 1 minute approximately to classify one news document. The time to classify the news documents can be reduced by implementing the feature selection methods. Apart from that, the accuracy of the classifier is hoped to be increased too.

In this research, it is found that the implementation of top- *n* feature selection affects accuracy of the classifier. By increasing the n value of top-*n*, the accuracy of the classifier is increasing too. In this experiment, the value of *n* is varied from *n* =10% to *n* =80%. The n value is the number of words in the news documents which is used as the representative keywords for the documents. As mentioned in chapter, this research applies top-n feature selection to reduce the number of feature in the document in order to enhance the classifier in term of accuracy and computation time. Thus for example, if the n value is set at *n* =10%, the classifier only use 10% of words with the highest weight in the document as the keywords to be representative of the documents. The results of category classification for various values of n are explained in Figure **4**.

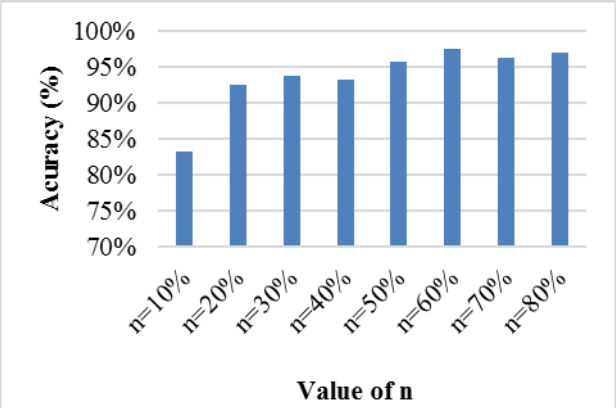


Figure 4 Result of Category Classification for Various n

The results show that the classifier achieved different accuracy by varying n value of top-*n*, with the lowest accuracy achieved at 83.13%, and the highest at 97.50%. The highest accuracy for the classifier is achieved when the n value is set at 60%. As seen in Figure 4.17, if the *n* value is set lower or bigger than 60%, the accuracy of the classifier became lower. This trend is occurred because of selecting fewer than 60% words as the keywords would yield a poorly described document, whilst more than 60% keywords would result in more noise.

In other hand, the increasing value of n has a disadvantage regarding the computation time. The classifier requires more time to classify the document due to the increasing of keywords for each document. The computation time for each n value of keywords is shown in Table **8**.

Table 8 Average Computational Time for Various n Values

	Computation Time
n=10%	5 seconds
n=20%	13 seconds
n=30%	19 seconds
n=40%	25 seconds
n=50%	30 seconds
n=60%	35 seconds
n=70%	40 seconds
n=80%	46 seconds

As seen in
Table **8**, the time to classify the document increases along with the increasing of n value. However, the performance of the classifier regarding the computation time can be considered as good as the required time to classify is less than 1 minute.

Apart from that, to see accuracy of the classifier for each news category, the best result of category classification when the n value is set at 60% is highlighted and illustrated in
Figure **5**.

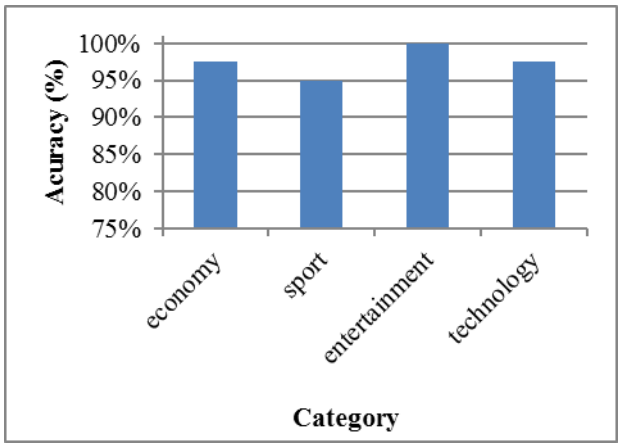


Figure 5 Result of Category Classification for Each Category

The results show that the classifier can perform well to identify the category, with the lowest accuracy achieved at 97.50%, and the highest at 100%. Albeit the high accuracy, the result also shows that the classifier performs differently as it process one category to another. Figure 4.18 shows that the classifier can produce a perfect accuracy for certain categories, while remaining relatively lower on others. This may be the result of the homogeneity of keywords which reside in a category. For example, category *Hiburan* (Entertainment) with 100% accuracy might have keywords which are very specific and self-explanatory such as song, movie, artist and so forth, which will not appear in any other category. In contrast, the category *Regional* at 95.00% accuracy level only comprises news because it does not own unique descriptors share similar keywords with other categories. However, the overall result of the category classification can be considered as good as the accuracy level for each category is beyond 90%.The average of the accuracy rate for the overall performance is summarized in
Table **9**.
It can be inferred from
Table **9** that the implementing of top-n feature selection method is successful to increase the performance of the classifier. The classifier achieved 97.50% as its best accuracy and took 35 seconds to classify the document when the n value is set at *n*=60% and the *k* value is 8. Even though time to classify the document increases along with the increasing of n value, the performance of the classifier regarding the computation time can be considered as good as the required time to classify is less than 1 minute.

Table 9 Summary of Category Classification Result

	Accuracy	Computation Time
n=10%	83.13%	5 seconds
n=20%	92.50%	13 seconds
n=30%	93.75%	19 seconds
n=40%	93.13%	25 seconds
n=50%	95.63%	30 seconds
n=60%	97.50%	35 seconds
n=70%	96.25%	40 seconds
n=80%	96.88%	46 seconds
n=100%	96.88%	52 seconds

Furthermore, to achieve the best accuracy by selecting *n* value at 60%, the classifier only requires 35 seconds approximately. This computation time is sufficient to be considered that the classifier has a good performance in term of speed to classify the news document. Thus, it can be concluded that the generic classifier can perform well to classify Indonesian and Malay news document by implementing the *k*-NN algorithm and top-*n* Feature selection method.

4.0 DISCUSSION

This presented *k*-NN algorithm and top-*n* feature selection method to perform category classification while a new automatic data collector and language identification algorithm was developed and then integrated into category classification. The integration of category classification, automatic data collector and language identification algorithm will create an integrated generic TC for Indonesian and Malay news document. There have been plenty of work focused on text classification, but only a few that are aimed at categorizing and identifying language, even less specifically for Indonesian and Malay news corpus. The approach shown in this thesis offers a method which is not just capable of classifying news into categories but is also capable of identifying the language. Further a new automatic data collector is developed to ease data collecting process. This effort is considered crucial; since there has been very limited number of approaches which

recognize the unique characteristics of the news domain, although such feature can severely affect the classifier's performance.

The *k*-NN algorithm is one most popular TC algorithm due to its low implementation cost and high degree of classification effectiveness [25, 42]. Furthermore, the technique has also been proven successful in working with news corpora difficulties. Nevertheless, the existing *k*-NN algorithm is only used to classify document for one language but none for similar language such Indonesian and Malay. In this thesis, the *k*-NN is employed to develop a generic TC for Indonesian and Malay news document. Modifications are required in order for the algorithm to be able to classify Indonesian and Malay news document. Prior to category classification process, language classification algorithm is employed. This language classification algorithm is assigned the news document into the correct language whether the news document belongs to Indonesian or Malay language

The methodology used in this research comprises of 2 principal stages: training and testing. The training stage basically prepares the classifier with training data before it starts classifying the testing set. In the training stage, online news documents are stored in the database as a corpus, and then pre-processed. Once it is pre-processed, the classifier selects the keywords and stores them back to the database. The top-*n* method is applied in the keywords selection step which takes place in both stages. Afterwards, the testing samples are classified in the testing phase. The keywords from the testing sample is selected and compared with the keywords from the database achieved in the training stage using *words computation* for the language identification and *cosine similarity calculation* for the category classification.

Experiments on Indonesian and Malay dataset have proven that the generic TC algorithms are able to identify the language and then classify Indonesian and Malay news documents. The Automatic Data collector algorithm is enable to retrieve the necessary information which is then used in language identification and category classification. This thesis produced a good result with an accuracy rate of up to 95.63% accuracy for language identification, and category classification for 97.50%. In terms of computational time, the results prove that language identifier works optimally by combining the stop words into words dictionary with an average of 5.52 seconds computational time while the category classifier works optimally on value of K = 8 and n = 60% with an average of 35 seconds computational time.

5.0 CONCLUSION

This presented k-NN algorithm and top-n feature selection method to perform category classification while a new automatic data collector and language identification algorithm was developed and then integrated into category classification. The integration of category classification, automatic data collector and language identification algorithm will create an integrated generic TC for Indonesian and Malay news document. There have been plenty of work focused on text classification, but only a few that are aimed at categorizing and identifying language, even less specifically for Indonesian and Malay news corpus. The approach shown in this thesis offers a method which is not just capable of classifying news into categories but is also capable of identifying the language. Further a new automatic data collector is developed to ease data collecting process. This effort is considered crucial; since there has been very limited number of approaches which recognize the unique characteristics of the news domain, although such feature can severely affect the classifier's performance.

References

[1] S. Brin and L. Page, 1998. The anatomy of a large-scale hypertextual Web search engine. *Comput. Netw. ISDN Syst.*, 30: 107-117.
[2] A. M. Z. Bidoki and N. Yazdani. 2008. DistanceRank: An intelligent ranking algorithm for web pages. *Inf. Process. Manage.* 44: 877-892.
[3] C. Tian. 2010. A Kind Of Algorithm For Page Ranking Based On Classified Tree In Search Engine, In *Computer Application And System Modeling (ICCASM). International Conference*. V13-538-V13-541.
[4] J. Elder IV and T. Hill. 2012. *Practical Text Mining And Statistical Analysis For Non-Structured Text Data Applications*: Academic Press, 2012.
[5] A. Hotho, A. Nürnberger, and G. Paaß. 2005. A Brief Survey of Text Mining, in *Ldv Forum*. 19-62.
[6] A. Kao and S. R. Poteet. 2007. *Natural Language Processing And Text Mining*: Springer Science & Business Media.
[7] F. Sebastiani, 2005. Text Categorization ed, 2005.
[8] B. Baharudin, L. H. Lee, and K. Khan. 2010. A Review Of Machine Learning Algorithms For Text-Documents Classification," *Journal Of Advances In Information Technology*. 1: 4-20.
[9] A. Kilgarriff and G. Grefenstette. 2003. Introduction To The Special Issue On The Web As Corpus, *Computational linguistics*. 29: 333-347.
[10] A. Selamat, H. Yanagimoto, and S. Omatu. 2002. Web news classification using neural networks based on PCA," in *SICE 2002. Proceedings of the 41st SICE Annual Conference*. 2389-2394.
[11] H. M. Noaman, S. Elmougy, A. Ghoneim, and T. Hamza. 2010. Naive Bayes classifier based Arabic document categorization," in *Informatics and Systems (INFOS), 2010 The 7th International Conference*. 1-5.
[12] S. Sakti, E. Kelana, H. Riza, S. Sakai, K. Markov, and S. Nakamura. 2008. Development of Indonesian Large Vocabulary Continuous Speech Recognition System within A-STAR Project," in *IJCNLP*. 19-24.

[13] T. Martin, T. Svendsen, and S. Sridharan. 2003. Cross-Lingual Pronunciation Modelling For Indonesian Speech Recognition. *Language*. 3: 2.
[14] A. D. Asy'arie and A. W. Pribadi. 2009. Automatic News Articles Classification In Indonesian Language By Using Naive Bayes Classifier Method, in *Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services*. 658-662.
[15] D. Y. Liliana, A. Hardianto, and M. Ridok. 2011. Indonesian News Classification using Support Vector Machine, *World Academy of Science, Engineering and Technology*. 57: 767-770, 2011.
[16] A. Firdan and K. E. Purnama. 2012. Classification of Emotions in Indonesian Texts using K-NN Method, *International Journal of Information and Electronics Engineering,* 2: 899-903.
[17] P. W. Buana, S. J. D.R.M., and I. K. G. D. Putra. 2012. Combination of K-Nearest Neighbor and K-Means based on Term Re-weighting for Classify Indonesian News, *International Journal of Computer Applications (0975 – 8887),* 50: 37-42.
[18] N. M. A. Lestari, I. K. G. D. Putra, and A. K. A. Cahyawan. 2013. Personality Types Classification for Indonesian Text in Partners Searching Website Using Naïve Bayes Methods, *IJCSI International Journal of Computer Science Issues*. 10: 1-8.
[19] S. Noah and F. Ismail. 2008. Automatic Classifications of Malay Proverbs Using Naive Bayesian Algorithm.
[20] H. Alshalabi, S. Tiun, N. Omar, and M. Albared. 2013. Experiments on the Use of Feature Selection and Machine Learning Methods in Automatic Malay Text Categorization. *Procedia Technology*. 11: 748-754.
[21] C. C. Aggarwal and C. Zhai. 2012. A survey of text classification algorithms, in *Mining text data*, ed: Springer, 163-222.
[22] A. McCallum and K. Nigam. 1998. A Comparison Of Event Models For Naive Bayes Text Classification, in *AAAI-98 workshop on learning for text categorization*. 41-48.
[23] P. Y. Pawar and S. Gawande. 2012. A Comparative Study On Different Types Of Approaches To Text Categorization, *International Journal of Machine Learning and Computing,* 2: 423-426.
[24] P. Y. Pawar and S. Gawande. 2011. A Comparative Study on Different Types of Approaches to Text Categorization, in *3rd International Conference on Machine Learning and Computing (ICMLC 2011)*. 366-369.
[25] V. Korde and C. N. Mahender. 2012. Text Classification And Classifiers: A survey, *International Journal of Artificial Intelligence & Applications (IJAIA).* 3: 85-99.
[26] L. H. Lee and D. Isa. 2010. Automatically Computed Document Dependent Weighting Factor Facility For Naïve Bayes classification, *Expert Systems with Applications,* 37: 8471-8478.
[27] K. A. Vidhya and G. Aghila. 2010. A Survey of Naive Bayes Machine Learning approach in Text Document Classification, *International Journal of Computer Science and Information Security (IJCSIS),* 7: 206-211.
[28] D. D. Lewis. 1998. Naive (Bayes) at forty: The Independence Assumption In Information Retrieval, in *Machine learning: ECML-98*, ed: Springer. 4-15.
[29] C. H. Lee and H. C. Yang. 2009. Construction Of Supervised And Unsupervised Learning Systems For Multilingual Text Categorization, *Expert Systems with Applications,* 36: 2400-2410.
[30] W. Zhang and F. Gao. 2011. An Improvement to Naive Bayes for Text Classification, *Procedia Engineering*. 15: 2160-2164.
[31] D. M. Farid, L. Zhang, C. M. Rahman, M. A. Hossain, and R. Strachan. 2014. Hybrid Decision Tree And Naïve Bayes Classifiers For Multi-Class Classification Tasks, *Expert Systems with Applications*. 41: 1937-1946.
[32] D. Li-guo, T. Taiyuan University of, D. Peng, and L. Ai-ping. 2014. A New Naive Bayes Text Classification Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering,* 12: 947-952.
[33] T. Joachims. 1998. *Text categorization with support vector machines: Learning with many relevant features*: Springer.
[34] M. Chau and H. Chen. 2008. A Machine Learning Approach To Web Page Filtering Using Content And Structure Analysis. *Decision Support Systems.* 44: 482-494.
[35] Y. Zhang, Y. Dang, H. Chen, M. Thurmond, and C. Larson. 2009. Automatic Online News Monitoring And Classification For Syndromic Surveillance. *Decision Support Systems*. 47: 508-517.
[36] O. Chapelle, V. Sindhwani, and S. S. Keerthi. 2008. Optimization Techniques For Semi-Supervised Support Vector Machines. *The Journal of Machine Learning Research*. 9: 203-233.
[37] V. T. Nguyen. 2010. Support Vector Machines Combined With Fuzzy C-Means For Text Classification. *IJCSNS*. 10: 222.
[38] Z. Wang, X. Sun, and D. Zhang. 2006. An Optimal Text Categorization Algorithm Based On SVM, in *Communications, Circuits and Systems Proceedings, 2006 International Conference.* 2137-2140.
[39] F. Colas and P. Brazdil. 2006. Comparison of SVM And Some Older Classification Algorithms In Text Classification Tasks, in *Artificial Intelligence in Theory and Practice*, ed: Springer. 169-178.
[40] A. Sun, E. P. Lim, and Y. Liu. 2009. On Strategies For Imbalanced Text Classification Using SVM: A Comparative Study, *Decision Support Systems.* 48: 191-201.
[41] Y. Yang and X. Liu. 1999. A Re-Examination Of Text Categorization Methods, in *Proceedings of the 22nd annual international ACM SIGIR Conference On Research And Development In Information Retrieval*. 42-49.
[42] S. C. Dharmadhikari, M. Ingle, and P. Kulkarni. 2011. Empirical Studies on Machine Learning Based Text Classification Algorithms, *Advanced Computing: An International Journal (ACIJ.* 2.
[43] E. H. S. Han, G. Karypis, and V. Kumar. 2001. *Text Categorization Using Weight Adjusted K-Nearest Neighbor Classification*: Springer.
[44] J. He, A.-H. Tan, and C.-L. Tan. 2003. On Machine Learning Methods For Chinese Document Categorization, *Applied Intelligence.* 18: 311-322.
[45] Q. Hu, D. Yu, and Z. Xie. 2008. Neighborhood classifiers, *Expert systems with applications.* 34: 866-876.
[46] X. Geng, T.-Y. Liu, T. Qin, A. Arnold, H. Li, and H. Y. Shum. 2008. Query Dependent Ranking Using K-Nearest Neighbor, in *Proceedings of the 31st Annual International ACM SIGIR Conference On Research And Development In Information Retrieval*. 115-122.
[47] J. Callut, K. Françoisse, M. Saerens, and P. Dupont. 2008. Semi-Supervised Classification From Discriminative Random Walks, in *Machine Learning and Knowledge Discovery in Databases*, ed: Springer. 162-177.
[48] R. Baeza-Yates and B. Ribeiro-Neto. 1999. *Modern Information Retrieval*. 463: ACM Press New York.
[49] Y. Yang and X. Liu. 1999. A Re-Examination Of Text Categorization Methods, Presented At The *Proceedings Of The 22nd Annual International ACM SIGIR Conference On Research And Development In Information Retrieval*, Berkeley, California, USA.