TEA: A Text Analysis Tool for the Intelligent Text Document Filtering

Jan Žižka[1], Aleš Bourek[2], and Luděk Frey[1]

[1] Department of Information Technologies
Faculty of Informatics, Masaryk University in Brno
Botanická 68a, 602 00 Brno, Czech Republic
{zizka, xfrey}@informatics.muni.cz
[2] Department of Biophysics
Faculty of Medicine, Masaryk University in Brno
Joštova 10, 662 43 Brno, Czech Republic
bourek@med.muni.cz

Abstract. This paper describes results achieved with a text-document classification tool TEA (TExt Analyzer) based on the naïve Bayes algorithm. TEA provides also a set of additional functions, which can assist users at fine-tuning the text classifiers and improving the classification accuracy, mainly through modifications of dictionaries generated during the training phase. Experiments, described in the paper, aimed at supporting work with medical unstructured text documents downloaded from the Internet. Good and stable results (around 97% of the classification accuracy) were achieved for selecting documents in a certain area of interest among a large number of documents from different areas.

1 Introduction

Nowadays, computer users can access incredibly large volumes of text documents via networks from databases and the Internet. Traditional retrieval methods, based on key words, typically provide as a response too many documents that are not actually relevant. Screening out uninteresting documents is usually a time-consuming and tedious work. Contemporary overview of selected Machine Learning tools to Information Retrieval can be found in [5]. Automated selection of interesting documents helps users to concentrate on their primary tasks. In the case of pure-text documents, an intelligent filter is entirely dependent on the word contents. The following sections describe a text analysis tool TEA, which is based on the naïve Bayes classifier and the task of which is to assist its users at training and using an effective text-document classifier.

2 The TEA Structure

TEA [2] consists of the following three parts: (1) the naïve Bayes classifier, (2) the training algorithm, and (3) supporting functions that enable users to work with word frequency dictionaries, to fine-tune the process of learning and classification, and to perform selected text analyses.

As the basic classification algorithm, TEA uses the naïve Bayes probabilistic classifier based on principles described in [6]. TEA learns from user-labeled sets of training examples. An example is a text document. Because the task of the classifier is to predict if a new, previously unseen example belongs to a group of interesting or uninteresting documents, two document classes are used: the *positive* class includes interesting documents while the *negative* class contains the rest. Each example is represented as a sequence of individual words (tokens). In the TEA system, a word is a string of characters delimited by white space. Before processing, all upper-case letters are converted into their lower-case equivalents. Strings longer than 15 characters are truncated.

The naïve Bayes classifier makes word independence assumptions which substantially reduce the number of parameters. Even if these assumptions are not generally correct, in practice it works well, see[1] or [4]. Let w_i stands for an i-th word in the dictionary containing all distinct words from the entire set of training examples. The degree to which a text document belongs to a certain class $c_j \in C = \{c_1, c_2, \ldots, c_{|C|}\}$, where $|C|$ is the number of classes, is given by the value $E(c_j)$:

$$E(c_j) = P(c_j) \prod_{i=1}^{n} P(w_i|c_j) \ , \tag{1}$$

$$P(c_j) = \frac{N_{docs_j}}{N_{docs}} \ , \tag{2}$$

where $E(c_j)$ is the estimate how much a new classified document is expected to be a member of c_j, $P(c_j)$ is an *a priori* probability that a document belongs to the class c_j, N_{docs_j} is the number of documents in the class c_j, N_{docs} is the total number of documents in all classes, $P(w_i|c_j)$ is the relative frequency (*a posteriori* probability) of w_i in the class c_j, and n is a number of words in the classified document. To avoid the problem with different number of interesting and uninteresting documents, it is useful to implement the m-estimate method for computations of terms $P(w_i|c_j)$:

$$P(w_i|c_j) = \frac{n_i^{(j)} + 1}{N_j + N_d} \ , \tag{3}$$

where $n_i^{(j)}$ is the occurrence number of the i-th word w_i in the class c_j, N_d is the total number of distinct words, and N_j is the number of distinct words in the class c_j (see [3]).

TEA's training process (see Table 1) assumes the existence of two classes of documents: one for positive (interesting, relevant) examples and one for negative (uninteresting, irrelevant) examples. The training is equivalent to setting up all the parameters that are needed for the classification process. Also, it is possible to retrain the classifier any time later when more examples are available. After training the naïve Bayes classifier, the user can utilize it to predict a class for new documents. The classification algorithm is straightforward (see Table 2).

Table 1. Algorithm 1: The training procedure.

1. For the entire set of N_{docs} training documents, store all the pairs *(a distinct word, its frequency)* into the dictionary.
2. For both possible classifications $c_j \in \{interesting,\ uninteresting\}$, and for all documents in the training set, find out N_{docs_j}, compute $P(c_j)$ and $P(w_i|c_j), i = 1, \ldots, N_d$.

Table 2. Algorithm 2: The classification procedure.

1. Within a new document, separate individual words w_i and find out their total number, n.
2. Compute the membership degree of the new document in each of the two classes c_j:
$$M_j = P(c_j) \prod_{i=1}^{n} P(w_i|c_j) \ . \tag{4}$$
3. Find out which class corresponds to the $\max(M_1, M_2)$, and assign it to the document.

In addition to the basic naïve Bayes method, TEA includes various functions that support the users' work with dictionaries that create the basis for the classifier. The dictionaries are built during the training process and contain words and their frequencies for the negative and positive examples. TEA's functions enable the users to customize classifiers to a high degree, depending on their area of interest; moreover, different users can share the same collection of text documents from different points of view. Another important task, which TEA supports, is the preprocessing of the raw text data obtained as the results of the users' queries. Unfortunately, the space available for describing the TEA system does not allow to provide full details; however, some essential TEA's functions are obvious from the following section, which describes experiments with medical text documents.

3 Results of Experiments

The experiments with training and classification were carried out to reveal possible strong and weak aspects of the naïve Bayes algorithm applied to real-world data obtained from the Internet. Particularly, extensive tests focused on pure-text documents—medical articles downloaded from MEDLINE using the conventional method based on sets of key words. Data obtained by this procedure typically contain many irrelevant and uninteresting documents, therefore the user cannot avoid some kind of postprocessing to get the needed information.

The results, described in this paper, were obtained using three series of tests. In all the cases, the classification of training examples was done manually by an expert in the field of interest. The first group of experiments aimed at finding documents relevant to the 'assisted reproduction.' Here, the set of interesting documents consisted of 701 articles while the uninteresting documents included topics such as 'contraception' (434 articles), 'gynecological endocrinology' (417) and 'computers in medicine' (258). Altogether, the set of uninteresting documents consisted of 1,109 articles from three different medical areas.

The goal of the second group of experiments was to find out how the classifier can assess which new documents are or are not interesting for the given user (provided that the same user classified training examples) in just one area of interest—in this case, 'assisted reproduction. ' The user marked 299 articles as interesting and 402 as uninteresting (together, 701 documents).

As the second group of experiments provided rather unstable results, the third group was carried out. In this case, the number of interesting documents was decreased by the random selection so that it was the same as the number of uninteresting documents, i.e., 299.

Each test series was performed four times and consisted of eight-fold cross-validation. Thus, each test was carried out 32 times. The total result was obtained as the average from the partial results. Generally, the classification was influenced by several factors that increased the difficulty of learning and decreased the classification accuracy. First, the documents contained a large number (12,631) of different word forms. Second, there were too many irrelevant attributes (words) which did not contribute to the classification at all, e.g., conjunctions, auxiliary verbs, etc. Third, the documents contained several kinds of noise: wrong classifications performed by the human expert, misspelled words, abbreviations having the form of a word, and so like. Some of the tests attempted to eliminate the influence of these factors.

3.1 Original Document Contents

The results of experiments with the naïve Bayes algorithm applied to the original data without any change of the word contents are shown in Table 3. In the column *Original*, the classification accuracy (always given in percentage) is shown for the original set of text documents. This set included 24 of 701 misclassified examples (i.e., examples which were incorrectly classified as uninteresting). After reclassification, the overall accuracy increased as indicates the column *Reclassified*. The fourth column, *Balanced*, shows the results obtained after balancing the number of interesting (299) and uninteresting (402) examples. Because of the decreased number of uninteresting examples (from 402 to 299), the accuracy slightly decreased, too. However, balancing the number of positive and negative examples resulted in the decreased standard deviation values, i.e., more stable results. Finally, the column *All Areas* shows the results of experiments that attempted to select interesting articles from one specific area ('assisted reproduction') among all topics mentioned above. Obviously, this was the strongest feature of the naïve Bayes algorithm. Here was the task of the classifier much easier because the interesting articles differed much more from the rest of the data unlike the other cases where the task was to find interesting documents just within one specific area. This was also the reason why the human expert originally misclassified some of the positive examples.

Table 3. The results of experiments with the original document contents. The numbers show the classification accuracy in percentage.

Original	Reclassified	All Areas	Balanced
68.0 ± 11.55	73.1 ± 11.54	94.3 ± 5.69	72.7 ± 8.04

3.2 Modified Document Contents

The main results are summarized in Table 4. The goal of this group of experiments was to find out how the classification accuracy and the stability of results could be improved by modifying the document contents using the tools included in TEA. From all the experiments, Table 4 shows the best results achieved. Essentially, the tests were divided into the three categories according to the method of selecting interesting articles: (1) within one area of interest—see the column *One Area*; (2) within documents with balanced number of positive and negative examples—see the column *Balanced*; and (3) within all medical areas covered by the documents—see the column *All Areas*. In addition, the document contents was gradually modified by: (1) removing the most common English words (100, 200, 300), including numbers (one, two, three, ...); (2) removing words with low occurrences (from 1 to 6) in the training set; (3) removing words with low rate between the positive and negative examples (if a certain word occurs with (almost)c the same—high or low—number in both classes, then its contribution to the classification can be neglected); (4) considering 'unknown' words, e.g., words in a foreign language—sometimes those words can contribute to a more accurate classification, so the user of TEA can assign different weights to such words; and (5) reducing the number of different word forms by using only their stem.

4 Conclusions

The naïve Bayes algorithm is an efficient tool for the document classification. The best results can be expected in the cases when the task is to select articles within a certain area of interest from a large number of different-topic documents. Once the articles are selected, the accuracy of a further, finer classification within a given area decreases. This paper suggest a tool which can, to a certain degree, improve results of the basic naïve Bayes algorithm by increasing or decreasing the influence of several text-document attributes. The experiments revealed that the

Table 4. The summarized results of experiments with the classification of medical text documents. The numbers show the classification accuracy in percentage.

	One Area	Balanced	All Areas
w/out the first 100 most common English words	73.6 ± 11.95	73.1 ± 8.09	95.4 ± 4.78
w/out words occurring less than twice	73.1 ± 12.38	72.6 ± 8.20	95.6 ± 4.57
w/out words having the low occurrence rate	70.1 ± 20.67	72.1 ± 10.51	97.1 ± 1.98
weighted unknown words	73.5 ± 8.34	71.9 ± 11.73	96.1 ± 3.95
using only the stem of words	71.1 ± 18.01	73.1 ± 9.63	95.4 ± 3.97

classification accuracy can be improved mainly by removing irrelevant attributes such as the first 100 most common words, words occurring in the documents less than twice or three times, and removing words which occur in interesting and uninteresting documents with (almost) the same ratio. These conclusions are not too surprising and are in keeping with other experiments carried out in the area of text-document retrieval. Tools like TEA can assist at the process of fine-tuning text document classifiers for individual users because the optimal setting is usually a difficult task.

The experiments with the real-world text documents also revealed that possibly the biggest disadvantage is the fact that the training needs both positive and negative examples. While it is relatively easy to find positive examples, it is practically impossible to find all negative examples as their number is usually too high. So the task of training strongly depends on creating a well-representative, balanced set of examples, which is, in practice, often impossible.

References

1. Domingos, P., Pazzani, M.: Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier. Proceedings of the 13^{rd} International Conference on Machine Learning ICML'96 (1996), 105-112.
2. Frey, L.: Automatic Filtration of Text Documents by machine Learning Methods. Master Thesis, Faculty of Informatics, Masaryk University in Brno (2000). In Czech.
3. Joachims, T.: A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. Proceedings of the 14^{th} International Conference on Machine Learning ICML'97 (1997), 143-151.
4. Lewis, D.D.: Naïve (Bayes) at Forty: The Independence Assumption in Information Retrieval. Proceedings of the 10^{th} European Conference on Machine Learning ECML'98. Springer Verlag Berlin Heidelberg New York (1998), 4-15.
5. Special Issue of Machine Learning Journal on Information Retrieval. Machine Learning Journal, Vol. 39, No. 2/3, May/June, 2000. Kluwer Academic Publishers (2000).
6. Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997).