Machine Learning Techniques for Homepage Detection

Ata Turk

Bilkent University, Department of Computer Engineering,
06800, Ankara, Turkey
atat@cs.bilkent.edu.tr

Abstract. This paper analyzes the efficiency and effectiveness of classical classifying methods in Machine Learning on the problem of discriminating personal homepages from regular Web pages. Two well-known classification methods are evaluated, which are kNN and naive Bayesian. We also applied a feature space dimensionality reduction technique depending on the document frequency and the probabilistic discriminative value of the features and calculated its effect on classification accuracy and time.

1 Introduction

Personal homepages are valuable information sources for describing the research interests of academicians. In a regular homepage of a researcher, you can probably find his CV, his list of publications, the courses he taught and he is currently teaching and most importantly his current research interests.

During the academic research process, learning the research issues and topics of other people working in your area can help greatly in producing solutions to the problems you have at hand and also it may be beneficial in identifying new research directions. Thus, a tool which lets you to explore homepages of people that work in the same area as you would be very beneficial.

Our focus, in this work, is on discriminating personal homepages from a collection of web pages. In other words, we are trying to mine homepages within an arbitrary set of Web pages. For this purpose, we analyze the effectiveness and efficiency of two classifying techniques on the homepage mining problem. The methods we study are kNN and naive Bayesian. As far as the author is concerned, there are no previous study regarding the identification of homepages, even though there are a number of studies on Web page categorization [1, **?**]. We believe that this study is the first step for a search engine which accepts resercher names as input and returns the homepages and names of other researchers working in similar topics with the input researcher as output.

The rest of the paper is organized as follows. In Section 2, existing classification methods that are known to be performing good in Web page categorization are analyzed. In Section 3, we provide some details about the classifiers we use and our dataset. In Section 4, experimental results are presented. Finally, we conclude in Section 5.

2 Classifiers and Feature Selection

We formally define the personal homepage detection problem as follows: Let D be a dataset that is composed of generic Web pages, and let d_t be a Web page within D. Then the personal homepage detection is learning from examples a function $f : d_t \rightarrow \{Homepage, Not - Homepage\}$ that maps each page d_t into one of *Homepage* or *Not-Homepage* classes. Thus, the problem is now a Web page categorization problem with two classes, which can be addressed with any one of the existing categorization algorithms in machine learning. For addressing this problem, we selected to use *k*NN and naive Bayesian categorization algorithms, both of which we describe below.

2.1 K-Nearest Neighbour (*k*NN) Classifiers

The *k-nearest neighbour* method has been proposed as a statistical pattern recognition approach by Dasarathy [3] in 1991 and it has been applied in text categorization by Masand et al. [4] in 1992.

In the *k-nearest neighbour* approach, in order to classify a new document, the system finds the *k-nearest neighbours* among the training documents and uses the categories of the *k-nearest neighbours* to decide on the category of the document [5]. There are a number of problems in the *k-nearest neighbour* approach. First of all, the number of comparisons required for the categorization of a single document is equal to the number of documents within the training set, which means that the *k*NN algorithm is not time and computation efficient, if the number of documents in the training set is very large. Secondly, the accuracy of the algorithm greatly depends on the similarity function used for determining the neighbours and the selection of an appropriate value for *k*. On the other hand, *k*NN algorithm is efficient for categorizing datasets with a large number of features and thus, it is being widely used in text categorization.

In personal homepage detection problem, features are the words that are extracted from documents and each document is represented as a feature vector. Hence, the dimension of feature space of the dataset is very large and applying *k*NN algorithm to this problem is appropriate. We investigate the effect of the size of the training data in the performance of the *k*NN classifier by conducting experiments with varying train data sizes.

2.2 Naive Bayesian Classifiers

From a probabilistic point of view, personal homepage detection can be thought as the estimation of the conditional probability $p(d_t) = P(C_t = Homepage|d_t)$. That is $f(d_t) = Homepage$ if $p(d_t) > 0.5$ and $f(d_t) = Not - Homepage$ otherwise. The naive Bayesian classifier computes $p(d_t)$ as

$$P(C_t = Homepage|d_t) = P(d_t|C_t = Homepage)P(C_t = Homepage). \quad (1)$$

In order to apply the naive Bayesian classifiers, we have to make the naive-Bayes assumption, which states that the probability of a word occurring in a

document is conditionally independent of the same probability for other words. Even though this assumption is wrong, experimental results prove that naive Bayesian classifiers that make this assumption still work well for text categorization. After this assumption, we can compute $P(d_t|C_t = Homepage)$ as

$$P(d_t|C_t = Homepage) = \prod_{i=1}^{|d_t|} P(w_t^i|C_t = Homepage). \quad (2)$$

where $|d_t|$ denotes the length of page d_t and w_t^i is the i-th word in page d_t.

The naive Bayes classifier, once trained, runs much faster than the kNN classifier and performs as good as the kNN if not better. The run-time of the naive Bayesian classifier can be effected from the dimension of the feature space. We investigate this issue by applying feature selection to reduce the feature space in our dataset.

Feature Selection for Naive Bayesian For reducing the dimension of the feature space, we propose an ad-hoc feature selection algorithms which takes the discriminative value of a feature into account. We describe the discriminative value of a feature w_i as the division of the number of homepages that w_i passes to the total number of documents that w_i passes 3.

$$Dv(w_i) = \frac{|Homepages - that - has - w_i|}{|documents - that - has - w_i|}. \quad (3)$$

Having calculated the $Dv(w_i)$ value for a feature, we keep that feature if its discriminative value is bigger than a *threshold* or lower than $1 - threshold$, i.e. $Dv(w_i) > thresholdor Dv(w_i) < (1 - threshold)$. In our calculations we find that 0.85 threshold value provides the highest improvement in accuracy of the naive Bayesian.

The experiments show that feature selection through discriminative value computation does not reduce the number of features significantly. Hence, we apply document frequency thresholding (DF) technique to further reduce the number of features. Document frequency is the number of documents in which a term occurs. We compute the $Df(w_i)$ value for each feature w_i in the dataset and remove the features which have a document frequency less than some predetermined *threshold*.

3 Experimental Settings

3.1 Classifier Properties

In our experiments, we used Cambazoglu's *Harbinger* [6] toolkit, which provides the implementation for some of the well-known and frequently used machine learning classifiers including naive Bayesian and k-nearest neighbour classifiers. The toolkit provides a wrapper program which allows you to set parameters such

Table 1. The number of pages coming from each university in the dataset.

Properties	Bilkent	Bosphorus	Waterloo	Penn. State	sum
Total number of pages	654	1470	949	877	3950
Number of homepages	565	242	501	537	1845
Number of other pages	89	1228	448	340	2105
Number of features					34435

as validation type and fold count on your data. Validation type can be one of cross-validation, shuffled-cross-validation, leave-1-out and all. If validation type is set to cross-validation or shuffled-cross validation the dataset is N-fold cross-validated (where N is the fold count). This is done by partitioning the data into N equal parts and running the classifier N times. In each run a different part of the dataset is used for testing. If validation type is set to all, the whole instance set is used for both training and testing. The *k*NN implementation of the Harbinger lets you specify the number of the neighbours to be found and voting metric to be used once the neighbours are determined. The voting metric can be one of majority voting or similarity voting.

3.2 Dataset Properties

Our dataset is composed of Web pages that have been crawled from the servers of the computer engineering departments of four different universities. These universities are: Bilkent University, Bosphorus University, Waterloo University and Pennsylvania State University. The number of pages collected from each university and some other important properties of the dataset are presented in Table 1.

The number of features in the dataset is equal to 34847 when we do not do any stop word elimination. We eliminate some common words in English such as *and, but, or* as they appear in almost all of the documents and hence, the number of features of our dataset reduces to 34435.

4 Experimental Results

The Harbinger toolkit is used for classifying the collected datasets. Whenever not stated explicitly, ten-fold shuffled cross validation is applied and the average results are announced.

Our first set of experiments test the effect of the dimension of the feature space on the performance of the naive Bayesian classifier and the results are depicted in Table 2 and Table 3. We can observe from Table 2 that reducing the number of features through selecting features according to their discriminative value increases the prediction accuracy and reduces the test time and training time of the classifier. However, when the threshold value exceeds 0.85, further

Table 2. The effect of feature selection with Discriminating Value method on the performance of naive Bayesian classifier.

Discr. Value	Num. of features	Train Time(sec.)	Test Time(sec.)	Pred. Accuracy(%)
—	34435	11.317	0.647	93.800
0.65	30856	9.986	0.580	94.200
0.70	29830	9.577	0.564	94.550
0.75	28914	9.218	0.537	94.850
0.80	28297	9.016	0.537	95.000
0.85	27602	8.891	0.525	95.100
0.90	26851	8.565	0.509	95.000
0.95	26509	8.512	0.493	93.800

Table 3. The effect of feature selection with Document Frequency thresholding (DF) on the performance of naive Bayesian classifier.

Min. DF	Num. of features	Train Time(sec.)	Test Time(sec.)	Pred. Accuracy(%)
2	20299	6.500	0.381	95.350
3	13717	4.391	0.278	93.950
4	11639	3.883	0.237	92.850
5	10630	3.385	0.219	91.600

reduction does not improve performance. For this dataset, threshold value of 0.85 as discriminating value is ideal. Unfortunately, discriminative value selection algorithm does not provide significant reductions in the number of features. Hence, we use document frequency (DF) method for further reduction in the feature size. The results of feature selection through DF are presented in Table 3. The best prediction accuracy is achieved when words which do not appear in more than one documents are not excepted as a feature. This is only natural as such features do not provide any value from categorization point of view. If the document that the feature passes is in the training set, there won't be any documents in the test set with that feature. On the other hand, if the document with the feature is in the test set, it will not be recognized by any of the training documents and hence will just cause noise in the classification process. One other observation is that if we increase the minimum DF value from two to three, the number of features reduces by 35%, with a cost of 1.5% reduction in prediction accuracy. This implies that if we can sacrifice from prediction accuracy, we can reduce test and training times significantly through DF feature selection algorithm.

Second set of experiments that we have conducted analyze the effect of the size of the training set and the number of neighbours on the prediction accuracy of kNN classification. In these experiments we have selected the distance-metric as euclidean distance and the voting-metric as majority voting. In 2-fold experiments half of the pages are tested using the other half as training set. In 10-fold experiments 10% of the documents are tested using the 90% of the documents

Table 4. The effect of the fold count and the number of nearest neighbours (k) on the performance of kNN classifier prediction accuracy.

	Prediction Accuracy(%)					
N-fold	k=3	k=5	k=7	k=9	k=11	k=13
2-fold	74.900	76.600	78.150	86.481	85.650	86.400
4-fold	81.350	82.600	83.900	87.063	89.050	86.127
6-fold	83.958	83.906	84.740	89.115	89.844	86.051
8-fold	84.323	84.427	84.531	87.646	86.886	91.615
10-fold	85.600	84.850	85.150	88.101	87.038	86.329

as training set. Our observation is that when we reduce the size of the training set, the prediction accuracy reduces as well. However, increasing the number of neighbours enhances the prediction accuracy and in fact, after 9 neighbours, the difference between 2-fold and 10-fold cross validation becomes negligible. As the test-time of the kNN greatly depends on the size of the training data, we observe that usage of high k values and small number of training pages provides good accuracy and low run-time cost values.

5 Conclusions

In this paper, we have investigated the performance of two well-known classification algorithms, namely the naive Bayesian and the kNN algorithms, on personal homepage detection problem. A simple comparison of the two algorithms imply that the naive Bayesian algorithm is more suitable for addressing this problem as it not only provides higher prediction accuracies but also runs much faster than the kNN algorithm.

During the analysis of the effect of the dimension of feature space to the run-time of the naive Bayesian classifier, we proposed an ad-hoc feature selection algorithm, discriminating value feature selection, which can improve prediction accuracy up to 1.5%. Unfortunately, the discriminating value feature selection method does not reduce the number of features significantly, but our experiments show that feature selection through document frequency thresholding can reduce the dimension of the document space significantly with a reduction in prediction accuracy.

Our investigations point to the usage of naive Bayes classifiers for personal homepage detection problem. As future study, we believe that a homepage detection system can be used for creating a hypergraph between researchers and their respective research areas. Utilizing that hypergraph, it can be possible to query for researchers which have the same research interests with a given researcher.

References

1. M.-Y. Kan, Web Page Categorization without the Web Page, Proceedings of the 13th International World Wide Web Conference, New York, Pages: 262 - 263, 2004.
2. A. Sun, E.P. Lim, and W.K. Ng. Web Classification Using Support Vector Machine. 4th International Workshop on Web Information and Data Management, Virginia, 2002.
3. B.V. Dasarathy, Nearest Neighbour (NN) Norms: NN Pattern Classification Techniques, IEEE Computer Society Press, Las Alamitos, CA, 1991.
4. B. Masand, G. Linoff, and D. Waltz Classifying new stories using memory based reasoning, Proceedings of 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM Press, New York 59-64, 1992.
5. C.D. Manning and H. Schutze Foundations of Statistical Natural Language Processing, MIT Press, Cambridge, MA, 1999.
6. http://www.cs.bilkent.edu.tr/ berkant/coding/harbinger