Application of k-Nearest Neighbor on Feature Projections Classifier to Text Categorization

Tuba Yavuz and H. Altay Guvenir

Department of Computer Engineering and Information Science
Bilkent University
06533 Ankara, Bilkent
{ytuba,guvenir}@cs.bilkent.edu.tr

Abstract. This paper presents the results of the application of an instance-based learning algorithm *k*-Nearest Neighbor Method on Feature Projections (*k*-NNFP) to text categorization and compares it with *k-Nearest Neighbor* Classifier (*k*-NN). *k*-NNFP is similar to *k*-NN except it finds the nearest neighbors according to each feature separately. Then it combines these predictions using a majority voting. This property causes *k*-NNFP to eliminate possible adverse effects of irrelevant features on the classification accuracy. Experimental evidence indicates that *k*-NNFP is superior to *k*-NN in terms of classification accuracy in the presence of irrelevant features in many real world domains.

1 Introduction

As technological improvements began to support the storage of high volume data, feasible implementation of applications that can use such amount of data came to discussion. Information Retrieval (IR) is such an area that the applications in its domain mostly require use of large amount of data. Moreover the documents that it deals with are mostly in natural language. Therefore high volume data is not the single factor that affects the design decisions for IR applications but also the content of the documents poses significant problems to deal with.

Text categorization, which is the process of assigning predefined categories to text documents, is just one of the hot topics of IR that requires flexibility to handle a large volume of data efficiently, to process and to understand the content of the data to a degree that will give meaningful results.

Many machine learning algorithms have been applied to text categorization so far. These include symbolic and statistical approaches. Experiments regarding these works give promising results. However, most of algorithms are not scaleable with the size of feature set, which is expressed in the order of tens of thousands. This requires reduction of feature set or training set in such a way that the accuracy would not degrade.

On the other hand, algorithms like *k*-NN [8] and Linear Least Squares Fit (LLSF) [9] mapping method can be used with large set of features compared to the other existing methods.

This paper examines the performance of a new version of the nearest neighbor algorithm, which is called *k*-NNFP [3], when applied to text categorization. The *k*-NNFP classifier is a variant of *k*-NN. The *k*-NNFP classifier finds the *k-Nearest Neighbors* separately for each feature whereas the *k*-NN classifier finds the *k-Nearest Neighbors* by considering all the features together. The experiment was done by using these classifiers to classify UseNet messages, from 20 different UseNet groups. Each message belongs to a single UseNet group, which is used as the category of the message (see Section 6). The results showed that k-NNFP could overcome the possible adverse effects of irrelevant features and achieves higher accuracy than k-NN in the case of even category distibution over the data set and comparable accuracy when there is an uneven category distribution over the data set.

The rest of the paper is organized as follows. In Section 2, results for previous work are given. In Section 3 and Section 4, k-NN and k-NNFP are explained in detail. Section 5 explains the application of k-NNFP to text categorization. In Section 6, our experimental setup is introduced and results of our experiments are given in Section 7. Finally Section 8 concludes the paper.

2 Previous Work

Research on application of nearest neighbor classification method to text categorization is rare. The ones that we are aware of are [1], [4] and [5].

Research done by Yang, which is the latest one among the others, gives promising results. Yang names his approach as ExpNet and uses a set of training instances that are assigned to categories by human experts. A network is constructed to reflect the conditional probabilities of categories to given texts. The network consists of three layers. The words, texts and categories are represented in the first, second and the third layers, respectively. The links between words and documents and the links between documents and categories show the frequency of the word in that document and the conditional probability of the category given that document respectively. The weights of the links are computed according to the statistics about word distribution and category distribution over the training set. ExpNet is used to find the relevancy ranking of the candidate categories. A cutoff point, which gives the best result, is found for the top-ranking categories.

ExpNet is evaluated on MEDLINE data set and tested with Linear Least Squares Fit (LSSF) mapping method for comparison. ExpNet showed a performance in Recall and Precision comparable to LSSF.

In another work by Yang [2], statistical approaches to text categorization were compared. Eleven methods (k-NN, LLSF, WH [10], RIPPER [11], EG [10], NNets [15], SWAP-1 [12], CHARADE [14], WORD (word matching), Rocchio [10], NaiveBayes [13]) were analyzed and k-NN was chosen as the performance baseline for several collections. On each collection performance results of other methods were normalized using the result of k-NN. Yang found out that k-NN is one of the top-performing classifiers. Also it was the only learning algorithm that has scaled to the full domain of MEDLINE categories.

3 The k-Nearest Neighbor Classifier

K-Nearest Neighbor classifier is an instance based learning method. It computes the similarity between the test instance and the training instances and considering the k top-ranking nearest instances, finds out the category that is most similar. There are two methods for finding the most similar instance: majority voting and similarity score summing.

In majority voting, a category gets one vote for each instance of that category in the set of k top-ranking nearest neighbors. Then the most similar category is the one that gets the highest amount of votes. In similarity score summing, each category gets a score equal to the sum of the similarity scores of the instances of that category in the k top-ranking neighbors. The most similar category is the one with the highest similarity score sum.

The similarity value between two instances is the distance between them based on a distance metric. Generally Euclidean distance metric is used.

Let an instance t with n features be represented with the feature vector

$$< v_1(t), v_2(t), ..., v_n(t) > \tag{1}$$

where $v_i(t)$ is the value of the ith feature of instance t. Then the distance between two instances t_i and t_j is $d(t_i, t_j)$, where

$$d(t_i, t_j) = \sqrt{\sum_{m=1}^{n} (v_m(t_i) - v_m(t_j))^2}. \tag{2}$$

Also this metric requires normalization of all feature values into the same range.

Inductive bias of k-NN method is that instances that are near to each other probably have similar categories. However, while the distance between two instances is calculated all the features are taken into account together and this raises a problem when the discriminative features are only a small subset of the whole feature set. In such a case even two instances that have identical values on these discriminative features may not be considered as near neighbors. Therefore accuracy of k-NN method is sensitive to the number of irrelevant features.

There are several ways to cope with this problem. One is proposed by Kelly and Davis [7]. It is the weighted k-NN method. A genetic algorithm is devised to learn the feature weights. After the assignment of weights to features, k-NN method is applied.

Applying feature selection to the data set before the categorization is another solution. Features that are irrelevant for the categorization task are first determined. Then k-NN algorithm is applied using only these relevant features. A comparative study of feature selection for text categorization can be found in [6]. Yet another solution is proposed by Guvenir and Akkus [3]. In the next section the proposed method, which is called k-NNFP (k-Nearest Neighbor on Feature Projections), will be introduced.

4 k-Nearest Neighbor on Feature Projections Classifier

The k-Nearest Neighbor on Feature Projections is a variant of k-NN method. The main difference is that instances are projected on their features in the n-dimensional space and distance between two instances is calculated according to a single feature. The distance between two instances t_i and t_j regarding feature m is $d_m(t_i, t_j)$, where

$$d_m(t_i, t_j) = v_m(t_i) - v_m(t_j). \tag{3}$$

Therefore, this metric does not require normalization of feature values. If there are f features, this method returns $f \times k$ votes whereas k-NN method returns k votes.

Since each feature is evaluated independently if the distribution of categories over the data set is even, votes returned for the irrelevant features will not adversely affect the final prediction.

If majority voting is used and the categories are evenly distributed over the test instances then the categories will also be evenly distributed in the returned votes and only the votes for relevant features will be effective. If the distribution of categories over the data set is not even then the irrelevant features will return the highest vote for the most frequently occurring category.

If similarity score summing is used and the categories are evenly distributed over the test instances then the similarity score sum of an irrelevant feature will be equal for each category and it will be not effective in the decision phase. However, if the categories are not evenly distributed then similarity score sum of an irrelevant feature will be higher for most frequently occurring class.

Therefore, k-NNFP cannot eliminate the adverse effect of irrelevant features if the category distribution is uneven over the training instances.

5 Application of k-NNFP to Text Categorization

As stated in Section 2, in [1] it is reported that k-NN method shows a performance in Recall and Precision comparable to LSSF mapping method, and significantly better then other methods tested. Also in a recent work of Yang [2], results of a comparative study for statistical approaches to text categorization shows that k-NN method is one of the top performing classifiers and it is the only method that has scaled to the full domain of MEDLINE categories.

According to the experiments reported in [3], k-NNFP method provides even better classification accuracy than k-NN method when a data set contains many irrelevant features on the artificially generated data sets and achieves comparable classification accuracy on real-word data sets.

Combining these facts about k-NN and k-NNFP and considering the reason for superiority of k-NNFP method over k-NN method, which is the ability to overcome the adverse effect of irrelevant features on classification accuracy, it is desirable to compare the two methods on a data set where irrelevant features frequently occurs. Text categorization is such an application area where data sets inevitably contain a high number of irrelevant features. Even after elimination of stop words, many irrelevant features still exist.

In text categorization, instances are natural language documents. An instance can be represented with a set of word and word weight pairs. There are many techniques for assigning weights to words; e.g. term frequency (TF), inverse document frequency (IDF) and $TF \times IDF$. Here

$$TF(word, doc) = \frac{\#\ of\ occurences\ of\ word\ in\ doc}{total\ \#\ of\ words\ in\ doc}. \tag{4}$$

If all the common terms of two documents are considered together, then similarity between two documents A and B is calculated as follows

$$sim(A, B) = \frac{\sum w_{tA} \times w_{tB}}{\|A\| \times \|B\|}. \tag{5}$$

If the similarity is computed on a single term (feature), then similarity between two documents A and B on a common term t is calculated as follows

$$sim(A, B) = w_{tA} \times w_{tB}. \tag{6}$$

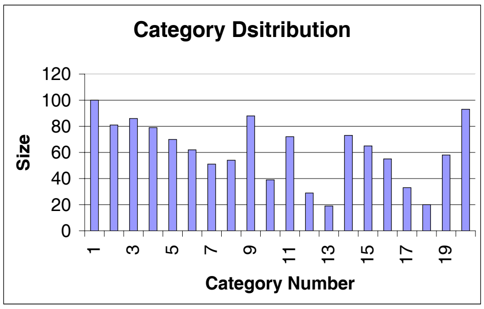


Figure 1: Data set with uneven category distribution

where

t is a word shared by A and B,
w_{tA} is the weight of word t in A,
w_{tB} is the weight of word t in B,
n is the number of terms in A,
m is the number of terms in B,
$\|A\|_2 = \sqrt{w_{1A}^2 + w_{2A}^2 + \ldots + w_{nA}^2}$ is the norm of A,
$\|B\|_2 = \sqrt{w_{1B}^2 + w_{2B}^2 + \ldots + w_{nB}^2}$ is the norm of B.

6 Experiments

In the experiments, we used UseNet messages as the documents to be classified. Mesages in this set were sent to a single newsgroup which is used as the category of the message. The data set contained 100 documents from each of 20 different newsgroups. The newsgroups used here are alt.atheism, sci.crypt, comp.graphics, sci.electronics, comp.os.ms-windows.misc, comp.sys.ibm.pc.hardware, sci.space, comp.windows.x, rec.sport.hockey, misc.forsale, talk.politics.guns, rec.autos, talk.politics.mideast, rec.motorcycles, sci.med, talk.politics.misc, rec.sport.baseball, comp.sys.mac.hardware, soc.religion.christianity, and talk.religion.misc.

A stop word list of 600 entries is used for stop word elimination. The stop word list used includes prepositions, articles and common verbs used in English. For the term weighting, TF is used. Accuracy is computed as

$$accuracy = \frac{\#\ of\ correct\ predictions}{\#\ of\ total\ predictions}. \tag{7}$$

Classification accuracy, using 5-fold cross-validation, for k values ranging from 1 to 10 is found. We had two choices (majority voting vs. similarity score summing) for final category prediction and two choices (even vs. uneven) for category distribution. Since k-NNFP copes with the adverse effect of irrelevant features when the category distribution is even, we wanted to find out whether k-NN could outperform k-NNFP or they had comparable performance in terms of prediction accuracy in such a situation. Therefore we conducted experiments both for even category distribution and uneven category distribution cases. To find out whether the final prediction method makes a difference in terms of accuracy, we did experiments where similarity score summing was used as the final category prediction method. Again both even and uneven category

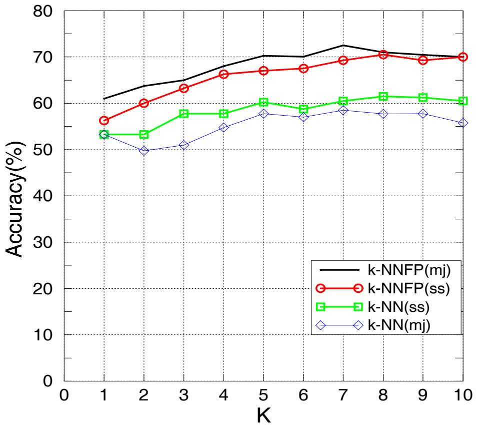


Figure 2: Accuracy results for even category distribution data set

distributions were considered. Data set with uneven category distribution was obtained by randomly removing different number of training instances for each category from the training set (see Figure 1). Therefore the training set contains different number of instances for each category. However, note that since the size of training set in the uneven data set is smaller than the case of even data set, both algorithms are expected to achieve lower accuracies in the uneven data set case than the even data set case. Consequently, four types of experiments ({majority voting vs. similarity score summing}{uneven vs. even category distribution}) were conducted. The main aims of these experiments were:

- Verifying the previously obtained results about the comparison of k-NNFP and k-NN in text categorization also.
- Observing the deficiency of k-NNFP in existence of irrelevant features and uneven category distribution.
- Comparing the results for majority voting and similarity score summing.

7 Results

As seen from Figure 2, previous observations on the comparison between k-NN and k-NNFP are verified also in the domain of text categorization. Given that categories are evenly distributed, k-NNFP achieves a better accuracy than k-NN.

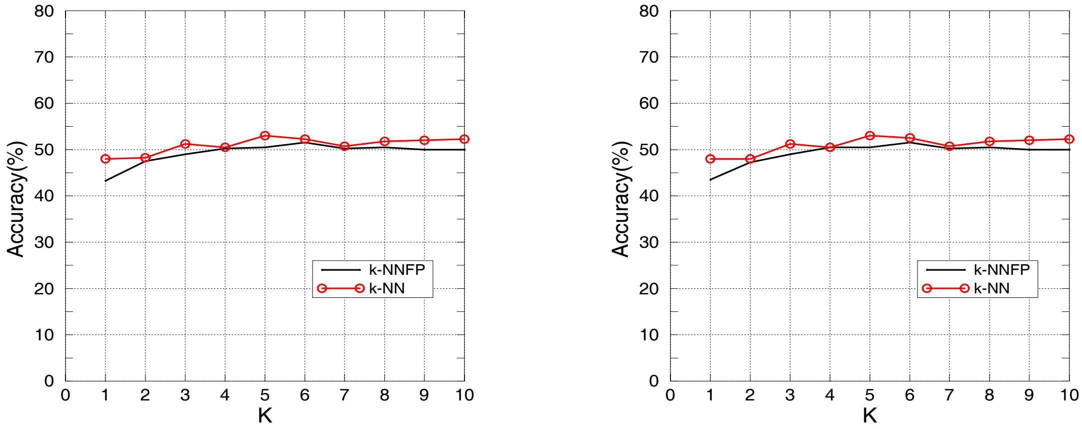
Figure 3(a) and Figure 3(b) indicate that when the categories are unevenly distributed and irrelevant features exist the accuracy k-NN achieves is slightly higher than k-NNFP.

Figure 2 shows the behaviors of k-NN and k-NNFP on majority voting and similarity score summing using even category distribution. As it is observed accuracy of k-NN is higher with similarity score summing than majority voting where as accuracy of k-NNFP is higher with majority voting than similarity score summing.

Figure 3(a), which shows the accuracy of k-NNFP and k-NN when categories are unevenly distributed and majority voting is used, is almost the same as Figure 3(b).

8 Conclusion

Text categorization is a hot topic since its application areas promise a lot for the information age. Since it deals with natural language documents there is some possibility



(a) (b)

Figure 3: Accuracy results for uneven category distribution data set: (a) Majority voting, (b) Similarity score summing.

that some amount of information will be lost since natural language is not interpreted successfully yet. At that stage we have to rely on the methods and heuristics developed in the machine learning area.

Machine learning has its own problems like irrelevant features, missing feature values, and scaling the methods to high volume of data. Therefore, in this research area, besides competing in terms of efficiency scientists are trying to find methods that will handle those problems of machine learning gracefully.

K-NN method is a good example for a learning algorithm that is comparably good in scalability and accuracy, but at the same time it has problems when irrelevant features exist in the data set. K-NNFP method, which is a variant of k-NN method, has the positive properties of k-NN method besides it can handle the irrelevant features gracefully.

In this paper, a new classification method called k-NNFP is applied to text categorization and compared with k-NN method in terms of classification accuracy. The weak property of k-NN which is the classification accuracy being adversely affected from the existence of irrelevant features is examined in comparison with k-NNFP method.

The aim of this paper, therefore, was to apply these two learning methods to text categorization, which is one of the most appropriate application areas that irrelevant features frequently occur in the data sets. 20 categories from UseNet newsgroups was chosen as the data set where each category contains 100 documents.

Results of the experiment indicate that theoretical expectation in elimination of irrelevant features for k-NNFP has also observed in text categorization besides some artificial data and real world data. The experiment done with an uneven category distribution shows that even the accuracy of k-NN in this situation is higher than k-NNFP the difference is not significant; k-NNFP have still comparable accuracy achievement. Also our results indicate that in the case of even category distribution, k-NN shows better performance with similarity score summing and for k-NNFP majority voting is more favorable. For both k-NN and k-NNFP, both final prediction methods work the same when the categories are unevenly distributed.

References

[1] Y. Yang. *Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorization and Retrieval.* In 17th Ann Int. ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94), 13-22, 1994.

[2] Y. Yang. *An Evaluation of Statistical Approaches to MEDLINE Indexing.* In Proceedings of the 1996 Annual Full Symposium of the American Medical Informatics Association (1996 AMIA), 358-362, 1996.

[3] A. Akkus and H. A. Guvenir. *K-Nearest Neighbor Classification on Feature Projections.* In Proceedings of ICML'96 Lorenza Saitta (ED.), Morgan Kaufmann, Bari, Italy, 12-19, 1996.

[4] R. Creecy, B. Masand, S. Smith and D. Waltz. *Trading Mips and Memory for Knowledge Engineering: Classify Census Returns on the Connection Machine.* Comm. ACM 35:48-63, 1992.

[5] B. Masand, G., Linoff, and D. Waltz. *Classifying News Stories Using Memory Based Reasoning.* In 15th Ann Int. ACM SIGIR Conference on Research and Development in Information retrieval (SIGIR 92), 59-64.

[6] Y. Yang and J.P. Pedersen *Feature selection in Statistical Learning of Text Categorization.* In Proceedings of the Fourteenth International Conference on Machine Learning, 1997.

[7] J. D. Kelly and L. Davis. *A Hybrid Genetic Algorithm for Classification.* In Proceedings of the Twelfth International Joint Conference on Artificial Intelligence, 645-650, 1991.

[8] S. Okamoto and K. Satoh. *An Average-Case Analysis of k-Nearest Neighbor Classifier.* In Proceedings of the First International Conference on Case-Based Reasoning , 243-264, 1995.

[9] Y. Yang and CG. Chute. *An Application of Least Squares Fit Mapping to Text Information Retrieval.* Proceedings of the sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval , 281-290, 1993.

[10] D. D. Lewis, R. E. Schapire, J. P. Callan, and R. Papka. *Training Algorithms for Linear Text Classifiers.* In SIGIR '96: Proceedings of the 19th annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 298-306, 1996.

[11] W. W. Cohen and Y. Singer. *Context-sensitive Learning Methods for Text Categorization.* In SIGIR '96: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 307-315, 1996.

[12] C. Apte, F. Damerau, and S. Weiss. *Towards Language Independent Automated Learning of Text Categorization Models.* In Proceedings of the 17th Annual ACM/SIGIR conference, 1994.

[13] D. D. Lewis and M. Ringuette. *Comparison of Two Learning Algorithms for text Categorization.* In Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94), 1994.

[14] I. Moulinier, G. Raskinis, and J. Ganascia. *Text Categorization: A Symbolic Approach.* In Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval, 1996.

[15] E. Wiener, J. O Pedersen, and A. S. weigend. *A Neural Network Approach to Topic Spotting.* In Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval (SDAIR'95), 1995.