An Adaptive Fuzzy kNN Text Classifier

Wenqian Shang[1], Houkuan Huang[1], Haibin Zhu[2], Yongmin Lin[1], Youli Qu[1], and Hongbin Dong[1]

[1] School of Computer and Information Technology, Beijing Jiaotong University, 100044, China
shangwenqian@hotmail.com
[2] Senior Member, IEEE, Dept. of Computer Science, Nipissing University, North Bay, ON P1B 8L7, Canada
haibinz@npissingu.ca

Abstract. In recent years, kNN algorithm is paid attention by many researchers and is proved one of the best text categorization algorithms. Text categorization is according to training set which is assigned class label to decide a new document which is not assigned class label belongs to some kind of document. Until now, kNN algorithm has still some issues to need to study further. Such as: improvement of decision rule; selection of k value; selection of dimensions (i.e. feature set selection); problems of multiclass text categorization; the algorithm's executive efficiency (time and space) etc. In this paper, we mainly focus on improvement of decision rule and dimension selection. We design an adaptive fuzzy kNN text classifier. Here the adaptive indicate the adaptive of dimension selection. The experiment results show that our algorithm is effective and feasible.

1 Introduction

With the development of the web, large numbers of documents are available on the Internet. Automatic text categorization becomes more and more important for dealing with massive data. It becomes a key technology to deal with and organize large numbers of documents. More and more methods based on statistical theory and machine learning has been applied to text categorization in recent years. For example, k-nearest neighbor (kNN)[1]-[4], Naive Bayes[5][6], Decision Tree[7][8], Support Vector Machines (SVM)[9], Linear least squares fit[10], neural network[11][12], SWAP-1, Rocchio and so on. Among these algorithms, kNN algorithm is studied by many researchers and is proved one of the best text categorization algorithms.

In recent years, many researchers study the improvement of kNN when the class distribution is uneven. They mainly focus on the improvement of decision function to resolve the problem of uneven class distribution, such as [13]-[17]. Our algorithm is based on the improvement of decision function too, but our improvement is very different from theirs. We adopt the theory of fuzzy sets, through analyzing the relationship among distance, similarity and membership function, according to the similarity, design a new weighted factor, at the same time we study the effect of dimension selection to

V.N. Alexandrov et al. (Eds.): ICCS 2006, Part III, LNCS 3993, pp. 216 – 223, 2006.

categorization performance. We design a formula of dimension selection. The experiment shows that our improvement is feasible.

2 The Classical kNN Algorithm Based on SWF Rule

At present, there are two main decision rules in kNN algorithm, that is, the discrete value rule DVF (Discrete-Valued Function) and the weighted similarity rule SWF (Similarity-Weighted Function). The most widely used is the SWF rule. This paper mainly focuses on this rule. The kNN algorithm based on SWF rule can be described as follows:

The system searches k documents (called neighbors) that have the maximal similarity (cosine similarity) in training sets. According to what classes these neighbors are affiliated with, it grades the test document's candidate classes. The similarity between the neighbor document and the test document is taken as this class weight of neighbor documents. The decision function can be defined as follows:

$$\mu_j(X) = \sum_{i=1}^{k} \mu_j(X_i) sim(X, X_i) \tag{1}$$

Where $\mu_j(X_i) \in \{0,1\}$ shows whether X_i belongs to ω_j ($\mu_j(X_i) = 1$ is True) or not ($\mu_j(X_i) = 0$ is False); where ω_j is the sort of document class; $sim(X, X_i)$ denotes the similarity between training document and test document. Then the decision rule is: If $\mu_j(X) = \max_i \mu_i(X)$, then $X \in \omega_j$.

3 The Improved kNN Decision Rule

In classical kNN algorithm, there is an obvious problem: when the density of training data is uneven it may decrease the precision of classification if we only consider the sequence of first k nearest neighbors but do not consider the differences of distances. To solve this problem, we adopt the theory of fuzzy sets, constructing a new membership function based on document similarities as follows:

$$\mu_j(X) = \frac{\sum_{i=1}^{k} \mu_j(X_i) sim(X, X_i) \frac{1}{(1 - sim(X, X_i))^{2/(b-1)}}}{\sum_{i=1}^{k} \frac{1}{(1 - sim(X, X_i))^{2/(b-1)}}} \tag{2}$$

Where j=1, 2, …, c, $\mu_j(X_i) sim(X, X_i)$ is the membership of known sample X to class j. If sample X belongs to class j then the value is 1, otherwise 0. From this formula, we can see that in reality the membership is using the different distance of every neighbor to the candidate classifying sample to weigh its effect. Parameter b is

used to adjust the degree of a distance weight. From paper [6] we can know that the best value field of b is between 1.5 and 2.5, always using 2, in this paper we take the value 2. Then fuzzy k-nearest neighbors' decision rule is: If $\mu_j(X) = \max_i \mu_i(X)$, then $X \in \omega_j$.

Why we amend formula (1) to formula (2), the reasons mainly are:

1) Similarity has relation to distance

A distance function or a similarity function is a general measurement of pattern recognition. In topology, a distance can be defined as [18]: suppose the space is Ω, x and y are arbitrary two points in this space. Mapping $d(x, y): \Omega \times \Omega \to R^+$ is called the two points' distance, if the mapping satisfies three conditions as follows:

(1) $d(x,x) = 0, \forall x \in \Omega$;
(2) $d(x, y) = d(y, x), \forall x, y \in \Omega$;
(3) $d(x, y) \le d(x, z) + d(z, y), \forall x, y, z \in \Omega$.

If the d(x, y)'s value field is [0, 1], i.e., $d(x, y): \Omega \times \Omega \to [0,1]$, then d(x, y) is called unitary distance between x and y and is expressed by $d_0(x, y)$.

We can use a unitary distance $d_0(x, y)$ to express similarity sim(x, y). Sim(x, y) can be defined as follows [19]: $f: d_0(x, y) \to sim(x, y) \in [0,1]$, where f must satisfy the following conditions:

If $d_0(x_1, y_1) > d_0(x_2, y_2)$, then $sim(x_1, y_1) \le sim(x_2, y_2)$;
If $d_0(x_1, y_1) < d_0(x_2, y_2)$, then $sim(x_1, y_1) \ge sim(x_2, y_2)$;
If $d_0(x_1, y_1) = d_0(x_2, y_2)$, then $sim(x_1, y_1) = sim(x_2, y_2)$.

Through the above definition, we can educe the function relationship between a distance and a similarity from the definition of distance, which is given independently; whereas we can educe another transfer relationship from the definition of distance, which is given by similarity. This is the internal relationship between a distance and a similarity. The relationship between a unitary distance $d_0(x, y)$ and a similarity sim(x, y) of arbitrary two points in feature space Ω can be described using complementary function $sim(x, y) = 1 - d_0(x, y)$ [19].

2) The similarity has relation to membership

The value field of a similarity and a membership is [0, 1]. This is not an occasion but a result of consanguineous affiliation between them. Through the concept of existing fields, we can relate a similarity with a membership. The existent field can be described as follows: around a point x in feature space Ω forms a field, for an arbitrary point y in the space, the value of this field can be measured by the similarity sim(x, y) that y towards x. Such a field is called an existing field. This field forms a fuzzy set

around the center point x, can be described as A. Any point in the field belongs to A. Its membership can be measured by the similarity between this point and the center point x. The nearer the distance is to x, the higher the possibility it belongs to A. Contrariwise it is not. The value of membership in the center point x is maximal, i.e., the constant value 1. Hence, from the meaning of existing field, around a point x in feature space Ω, there exists a potential fuzzy set A, the similarity which any point y for A in this set can be measured by the similarity between y and x, that is, $\mu_A(y) = sim(x, y)$ [19].

4 The Research of Dimension Selection

In text categorization, dimension is defined as the number of the feature words in VSM (Vector Space Model). For text documents, the dimension is always thousands upon thousands. Such high dimension is not permitted by a classifier. This is so called curse of dimensionality. Among this feature words, only a lot is useful for text categorization, many of them are noise words which hurt the performance of text categorization.

At present, there are many methods to reduce the dimension space. In text categorization, people always use feature selection method to reduce the dimension, such as Information Gain, Cross Entropy, Mutual Information, CHI, Weight Evidence of Text, Odds Ratio and so on. These methods can reduce the dimension space greatly. But there still over thousands feature words after feature selection. So how many dimensions to be selected are proper; how many dimensions to be selected are to make the categorization performance best.

After we study other authors' experiments in kNN and its variants and the research of our own experiment, we find that when the classes and number of documents in training set are certain we can approximately make sure the dimensions of selection. It follows the formula as follows:

$$\dim ensions = \left\lfloor \frac{\lfloor \log(num(\max(class))) \rfloor}{\lceil \ln(num(\min(class))) \rceil} \right\rfloor \times 1000 \tag{3}$$

Where num(max(class)) is the document numbers of the maximum class in training set, num(min(class)) is the document numbers of the minimum class in training set. If $\left\lfloor \frac{\lfloor \log(num(\max(class))) \rfloor}{\lceil \ln(num(\min(class))) \rceil} \right\rfloor$ is less than 1, we consider it as 1.

5 Experiment

5.1 The Datasets

In this paper, we use two datasets to validate our algorithm. One dataset comes from the International Database Center, Dept. of Computing and Information Technology,

Fudan University, China. The other dataset is Reuters-21578. We select 15 classes among 90 classes. The distribution of the class is uneven.

5.2 Experiment Result and Analysis

In experiment of Fudan's dataset, the feature dimension is 2000; the step of k is 5. In experiment of Reuters-21578, the feature dimension is 1000; the step of k is 5 too. The experiment result can be described as Table 1 and Table 2:

Table 1. The categorization performance in Chinese dataset when k is different

Value k	kNN		fkNN	
	Macro-F1	Micro-F1	Macro-F1	Micro-F1
10	81.972	79.907	83.802	82.346
15	81.571	79.326	83.441	81.882
20	81.077	78.397	82.564	80.720
25	80.717	77.468	82.547	80.372
30	80.307	76.887	81.876	79.443
35	80.061	76.423	81.627	78.978
40	78.458	74.100	81.016	78.165
45	77.885	74.100	80.612	77.700
50	77.657	72.822	80.232	77.120

Table 2. The categorization performance in Reuters-21578 dataset when k is different

Value k	kNN		fkNN	
	Macro-F1	Micro-F1	Macro-F1	Micro-F1
10	67.000	86.664	68.529	86.483
15	66.855	86.628	69.472	86.991
20	66.715	86.737	69.179	86.919
25	66.942	85.519	69.560	87.028
30	66.540	86.337	69.499	86.882
35	65.738	86.374	68.300	86.810
40	65.033	86.374	68.050	86.919
45	64.446	86.337	68.364	87.100
50	60.739	86.265	67.827	86.919

From Table 1 and Table 2, we can find that the improved kNN algorithm (fkNN) show better categorization performance than kNN no matter what the dataset is Chinese data or Reuters-21578. It has about 3% improvement in average performance. This proves that our improvement in decision rule is effective and feasible. This method solves the uneven problem of class distribution better.

Table 3. The categorization performance in Chinese dataset when dimension is different. Note that the highest accuracy is highlighted with bold font

dimension	kNN		fkNN	
	Macro-F1	Micro-F1	Macro-F1	Micro-F1
1000	**80.207**	**76.655**	**82.133**	**79.791**
2000	77.735	73.287	80.612	77.700
3000	78.219	73.287	81.931	79.210
4000	76.150	70.151	80.155	76.423
5000	76.396	70.267	79.306	74.913
6000	74.671	67.596	77.440	72.358
7000	69.258	60.163	74.595	67.712
8000	76.150	70.151	70.527	62.137

Table 4. The categorization performance in Reuters-21578 dataset when dimension is different. Note that the highest accuracy is highlighted with bold font

dimension	kNN		fkNN	
	Macro-F1	Micro-F1	Macro-F1	Micro-F1
1000	64.446	**86.337**	**68.364**	**87.100**
2000	**64.473**	84.375	67.773	84.847
3000	63.525	84.012	67.675	84.375
4000	63.977	83.830	65.696	84.230
5000	63.177	83.539	65.755	84.230
6000	63.250	83.503	65.728	84.048
7000	63.412	83.612	65.623	84.121
8000	64.030	83.576	65.853	83.975
9000	63.209	83.321	65.751	83.939
10000	62.829	83.031	65.896	83.830

Note that in Table 3 and Table 4, k takes 45. From Table 5, we can see that when dimension is 1000, the categorization performance of kNN and fkNN reaches the best. This result is very accordant with our dimension design formula: $\left\lfloor \frac{\lfloor \log(619) \rfloor}{\lceil \ln(59) \rceil} \right\rfloor \times 1000 = 1 \times 1000 = 1000$. From Table 6, we can find that when dimension is 1000, kNN and fkNN's categorization performance is the best. This result is consistent with our dimension formula design too. We use our dimension formula design in other author's kNN and its variant experiment [13][14][21]. It has the same good result of fitting our dimension formula. Even though it can not get the best point, it must be the better point. The experiment result shows that this dimension formula design is effective and feasible.

6 Conclusion

In this paper, we mainly discuss the improvement of decision rule and design a new algorithm of fkNN (fuzz kNN) to improve categorization performance when the class distribution is uneven. Based on this, we study the selection of dimensions and design a dimension selection formula. The experiment proves that our method is effective.

In the future, we need to study further on how to select the k; the impact of value k to dimension selection; how to improve the decision rule further, what their effects to be on each other and so on.

Acknowledgment

This research is partly supported by Beijing Jiaotong University Science Foundation under the grant 2004RC008.

References

1. Cover, T. M., Hart P. E.: Nearest neighbor pattern classification. IEEE Transaction on Information Theory. Vol. IT-13 (1967) 21-27, 1967
2. Yang, Y.: An Evaluation of Statistical Approaches to Text Categorization. Information Retrieval. Vol. 1 (1997) 76-88
3. Yang, Y., Lin X.: A Re-examination of Text Categorization Methods. In: Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in the Information Retrieval. ACM Press, New York (1999) 42-49
4. Masand, B., Lino, G., Waltz, D.: Classifying news stories using memory based reasoning. In: 15th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval. Copenhagen (1992) 59-64
5. Lewis, D. D.: Naïve (Bayes) at forty: the independence assumption in information retrieval. In: Proc. of the 10th European Conference on Machine Learning. Chemnitz (1998) 4-15
6. Mccallum, A., Nigam K.: A comparison of event models for naïve bayes text classification. In: AAAI-98 Workshop on Learning for Text Categorization. Madison, Wisconsin (1998) 41-48
7. Lewis, D. D., Ringuette M.: Comparison of two learning algorithms for text categorization. In: Proc. of the Third Annual Symposium on Document Analysis and Information Retrieval. Las Vegas (1994) 81-93
8. Apte, C., Damerau, F., Weiss, S.: Text mining with decision rules and decision trees. In: Proc. of the Conference on Automated Learning and Discovery, Workshop 6: Learning from Text and the Web. CMU (1998) 487-499
9. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Proc. of the 10th European Conference on Machine Learning. Chemnitz (1998) 137-142
10. Yang, Y., Chute, C. G.: An example-based mapping method for text categorization and retrieval. ACM Transaction on Information System. Vol. 12 (1994) 252-277
11. Ng, H. T., Goh, W. B., Low, K. L.: Feature selection, perceptron learning, and a usability case study for text categorization. In: 20th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval. (1997) 67-73

12. Wiener, E., Pedersen, J. O., Weigend, A. S.: A neural network approach to topic spotting. In: Proc. of the 4th Annual Symposium on Document Analysis and Information Retrieval. (1995) 317-332
13. Tan, S.: Neighbor-weighted K-nearest neighbor for unbalanced text corpus. Expert Systems with Application. Vol. 28 (2005) 667-671
14. Han, E., Karypis, G., Kumar, V.: Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification. In: Proc. of 5th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining. (2001) 53-66
15. Shankar, S., Karpis, G.: A Feature Weight Adjustment Algorithm for Document Categorization. In: Proc. of the International Workshop on Multimedia Data Mining. (2000)
16. Li, B., Lu, Q., Yu, S.: An Adaptive k-Nearest Neighbor Text Categorization Strategy. ACM Transactions on Asian Language Information Processing. Vol. 3 (2004) 215-226
17. Lim, H.: An Improved KNN Learning Based Korean Text Classifier with Heuristic Information. In: Proc. of the 9th International Conference on Neural Information Processing. (2202) 731-735
18. Dubois, D., Prade, H.: Fuzzy sets and systems (Theory and application). Oxford, Uk, Academic Press. (1980)
19. Zhao, S.: The method of fuzzy mathematics in pattern recognition. School of the West-North Electronic Engineering Press, Xi'an. (1987)
20. Bian, J., Zhang, X.: Pattern recognition. Tsinghua University Press, Beijing (2000)
21. Cardoso-Cachopo, A., Oliveira, A. L.: An empirical comparison of text categorization methods. In Proceedings of the 10th International Symposium on String Processing and Information Retrieval (SPIRE'03), number 2857 in Lecture Notes in Computer Science, Springer Verlag. (2003) 183-196