Combining Multiple K-Nearest Neighbor Classifiers for Text Classification by Reducts

Yongguang Bao and Naohiro Ishii

Department of Intelligence and Computer Science, Nagoya Institute of Technology, Nagoya, 466-8555, Japan
{baoyg, ishii}@egg.ics.nitech.ac.jp

Abstract. The basic k-nearest neighbor classifier works well in text classification. However, improving performance of the classifier is still attractive. Combining multiple classifiers is an effective technique for improving accuracy. There are many general combining algorithms, such as Bagging, or Boosting that significantly improve the classifier such as decision trees, rule learners, or neural networks. Unfortunately, these combining methods do not improve the nearest neighbor classifiers. In this paper we present a new approach to general multiple reducts based on rough sets theory, in which we apply multiple reducts to improve the performance of the k-nearest neighbor classifier. This paper describes the proposed technique and provides experimental results.

1 Introduction

As the volume of information available on the Internet and corporative intranets continues to increase, there is a growing need for tools finding, filtering, and managing these resources. The purpose of text classification is to classify text documents into classes automatically based on their contents, and therefore plays an important role in many information management tasks. A number of statistical text learning algorithms and machine learning techniques have been applied to text classification. These text classification algorithms have been used to automatically catalog news articles [1] and web pages [2], learn the reading interests of users [3], and sort electronic mails [4].

The basic k-nearest neighbor (kNN) is one of the simplest methods for classification. It is intuitive, and easy to understand, provides good generalization accuracy for text classification. Recently, researchers have begun paying attention to combining a set of individual classifiers, also known as a multiple model or ensemble approach, with the hope of improving the overall classification accuracy. Unfortunately, many combining methods such as Bagging, Boosting, or Error Correcting Output Coding, do not improve the kNN classifier at all. Alternatively, Bay [6] has proposed MFS, a method of combining kNN classifiers using multiple features subsets. However, Bay has not described a certain way for selecting the features, in other words, MFS should be built by trial and error. To overcome this weakness, Itqon *et al.* [7] use the test features instead of

S. Lange, K. Satoh, and C.H. Smith (Eds.): DS 2002, LNCS 2534, pp. 340–347, 2002.

the random features subsets to combine multiple kNN classifiers. However, the complexity of computing all test features is NP-hard, and the algorithm of computing test features in [7] is infeasible when the number of features is large, for example, to text classification problem.

In this paper, we present RkNN, an attempt of combing multiple kNN classifiers using reducts instead of the random feature subsets or the test features. A reduct is the essential part of an information system that can discern all objects discernible by the original information system. Furthermore the multiple reducts can be formulated precisely and in a unified way within the framework of Rough Sets theory. This paper proposes a hybrid technique using Rough Set theory to generate multiple reducts. Then these multiple reducts are used to improve the performance of the k-nearest neighbor classifier.

2 Information Systems and Rough Sets

2.1 Information Systems

An information system is composed of a 4-tuple as follow:

$$S =< U, Q, V, f >$$

Where U is the closed universe, a finite nonempty set of N objects $(x_1, x_2, ..., x_N)$, Q is a finite nonempty set of n features $\{q_1, q_2, ..., q_n\}, V = \bigcup_{q \in Q} V_q$, where V_q is a domain(value) of the feature $q, f : U \times Q \to V$ is the total decision function called the information such that $f(x, q) \in V_q$, for every $q \in Q$, $x \in U$.

Any subset P of Q determines a binary relation on U, which will be called an indiscernibility relation denoted by $INP(P)$, and defined as follows: $xI_B y$ if and only if $f(x, a) = f(y, a)$ for every $a \in P$. Obviously $INP(P)$ is an equivalence relation. The family of all equivalence classes of $INP(P)$ will be denoted by $U/INP(P)$ or simply U/P; an equivalence class of $INP(P)$ containing x will be denoted by $P(x)$ or $[x]_p$.

2.2 Reduct

Reduct is a fundamental concept of rough sets. A reduct is the essential part of an information system that can discern all objects discernible by the original information system.

Let $q \in Q$. A feature q is *dispensable* in S, if $IND(Q - q) = IND(Q)$; otherwise feature q is *indispensable* in S.

If q is an indispensable feature, deleting it from S will cause S to be inconsistent. Otherwise, q can be deleted from S.

The set $R \subseteq Q$ of feature will be called a *reduct* of Q, if $IND(R) = IND(Q)$ and all features of R are indispensable in S. We denoted it as $RED(Q)$ or $RED(S)$.