Expert Systems with Applications 39 (2012) 765–772

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Text categorization algorithms using semantic approaches, corpus-based thesaurus and WordNet

Cheng Hua Li, Ju Cheng Yang, Soon Cheol Park *

Department of Mathematics, Statistics and Computer Science, St. Francis Xavier University, Antigonish, Nova Scotia, Canada B2G 2W5
School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, Jiangxi 330013, PR China
Division of Computer Engineering, Chonbuk National University, Jeonju, Jeonbuk 561-756, Republic of Korea

ARTICLE INFO

Keywords:
Text categorization
Corpus-based thesaurus
WordNet
k-NN
BPNN
Neural network

ABSTRACT

In this paper, a corpus-based thesaurus and WordNet were used to improve text categorization performance. We employed the *k*-NN algorithm and the back propagation neural network (BPNN) algorithms as the classifiers. The *k*-NN is a simple and famous approach for categorization, and the BPNNs has been widely used in the categorization and pattern recognition fields. However the standard BPNN has some generally acknowledged limitations, such as a slow training speed and can be easily trapped into a local minimum. To alleviate the problems of the standard BPNN, two modified versions, Morbidity neurons Rectified BPNN (MRBP) and Learning Phase Evaluation BPNN (LPEBP), were considered and applied to the text categorization. We conducted the experiments on both the standard reuter-21578 data set and the 20 Newsgroups data set. Experimental results showed that our proposed methods achieved high categorization effectiveness as measured by the precision, recall and *F*-measure protocols.

1. Introduction

Automatic text categorization has been extensively used both in the natural language process and in the organization and management of information. This categorization has gained a prominent status in information systems and the data mining field due to the increased availability of documents in digital formats and the ensuing need to organize them. Many machine learning algorithms, including the Rocchio algorithm (Salton, 1971), support vector machine (Joachims, Nedellec, & Rouveirol, 1998), *k*-Nearest Neighbor algorithm (*k*-NN) (Han, Karypis, & Kumar, 2001; Li, Yu, & Qin, 2003; Tan, 2006), neural networks (Ruiz & Srinivasan, 1999), decision trees (Cohen & Singer, 1999), inductive rule learning (Apťe, Damerau, & Weiss, 1994), have been applied to text categorization tasks.

The *k*-NN is a similarity or distance based learning algorithm that has been shown to be very effective for a variety of the problem domains including text categorization. This approach has become a standard within text categorization and is included in numerous experiments as a basis for comparison. The *k*-NN takes an arbitrary input document and ranks the *k* nearest neighbors among the training documents through the use of a similarity score. *k*-NN then adapts the category of the most similar document or documents. As the documents can have more than one category assigned to them, we will try a number of ways of selecting the categories from the *k*-NN. Since the original *k*-NN algorithm is simple but efficient for text categorization, many modified and improved methods based on the *k*-NN approach have been proposed (Li, Yu, & Qin, 2003; Tan, 2006).

Many different neural networks have been also used as the classifiers for text categorization. Perceptron is the earliest and simplest form of neural network; it consists of only an input and an output layer. Ng, Goh, and Low (1997) first applied it and reported that it showed a surprisingly high performance. Zhang and Zhou (2006) proposed a multi-label neural network and its application to text categorization. Nonlinear neural networks are more sophisticated, with some hidden layers between the input and output layers (Ruiz & Srinivasan, 1998). The most successful method for supervised learning in neural networks is the back-propagation algorithm which was introduced by Rumelhart, Durbin, Goldenand, and Chauvin (1995) and Rumelhart and McClelland (1986). Ruiz and Srinivasan (1998) compared the categorization results using a back propagation learning mechanism and a counter-propagation learning mechanism.

There are many previous works that have done on automatic thesaurus generation for information retrieval. Qiu and Frei (1993) worked on a term by term similarity matrix based on how the terms of the collection are indexed. Zazo, Figuerola,

* Corresponding author at: Division of Computer Engineering, Chonbuk National University, Jeonju, Jeonbuk 561-756, Republic of Korea. Tel.: +82 63 270 2467; fax: +82 63 270 2461.
E-mail addresses: lchjk@msn.com (C.H. Li), yangjucheng@gmail.com (J.C. Yang), scpark@chonbuk.ac.kr (S.C. Park).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.07.070

Berrocal, and Rodriguez (2005) developed a work using similarity thesauri for Spanish documents. Perez-Aguera and Araujo (2006) shows how to use handmade thesauri combined with statistical methods to automatically generate a new thesaurus for a particular knowledge domain. The other thesaurus presented here is WordNet (WN) (Miller, Beckwidth, Fellbaum, Gross, & Miller, 1990) which is one of the most successful hand-crafted thesauri. WN is available in a machine readable form. It means that WN is ready to be used with other computer programs easily. However, some kinds of words are not included in WN, such as numbers, simplified terms and proper names. To overcome the lacks in this paper, the corpus-based automatically constructed thesaurus was combined to WN as Perez-Aguera and Araujo (2006) did.

This paper was to experiment the *k*-NN algorithm and the back propagation neural network algorithms in order to achieve an advanced result of text categorization. Furthermore, both the automatically constructed thesaurus and WN were incorporated into the categorizing process (Bang, Yang, & Yang, 2006). The automatically constructed thesaurus consists of a set of weighted term associations that are based on the hypothesis that terms have relationships if they co-occur often in documents (Xu & Croft, 1996).

The rest of paper was organized as follows. The *k*-NN and BPNN algorithms were described in Section 2. The thesauri, including the corpus-based thesaurus (CBT) and WordNet (WN) were described in Section 3. The experimental results were analyzed in Section 4. Conclusions were given in Section 5.

2. Categorization algorithms

2.1. k-Nearest Neighbor algorithm

The *k*-Nearest Neighbor algorithm (*k*-NN) is a method for classifying objects based on the closest training examples in the feature space. It is one of the most fundamental and simple categorization methods and should be one of the first choices for a categorization study when there is little or no prior knowledge about the distribution of the data.

The *k*-NN for the categorization is simple. Given example y to be classified, find the k examples (or neighbors) in the training data that are closest to it. If most of these neighbors belong to class I, then y is assigned to class I. The *k*-NN can be viewed as a voting system in which some neighbors vote for class I and others vote for class J. The major steps of the *k*-NN algorithm can be summarized as follows:

Training:

1. Construct the term by document training matrix D, where each row corresponds to a training document, and each column represents a word.
2. For each training example $\langle x, c(x)\rangle$ that belongs to D, compute the corresponding term weight of document vector d_x for document x.
3. Do a feature selection by choosing the features with the highest term weight.
4. Construct the thesaurus using the selected features in the D matrix.
5. Generate the new feature vectors for training documents using the final feature weighting method.

Testing y:

6. For each $\langle y, c(y)\rangle$ that belongs to D, compute the corresponding term weight of document vector d_y for document y.
7. Do a feature selection by choosing the features with the highest term weight.
8. Generate the new feature vectors for test documents using the final feature weighting method.
9. Calculate the cosine similarity between the test example and each training example $S_x = \cos Sim(d_y, d_x)$.
10. Sort examples x in D by decreasing the value of S_x.
11. Let N be the first k examples in D (get the most similar neighbors).
12. Return the majority class of examples in N.

2.2. Back propagation neural network algorithms

The back propagation neural network (BPNN) algorithm is the most popular of the neural network applications. The topology structure of the standard BPNN algorithm is shown in Fig. 1. There is an input layer, an output layer, and one or more hidden layers in this network. During training, the network is given an input pattern applied to the input layer. Based on the given input pattern, the network will compute the output in the output layer. This network output is then compared with the desired output pattern. The aim of the back-propagation learning rule is to define a method to adjust the weights of the networks. Eventually the network will give an output that matches the desired output pattern given any input pattern in the training set.

The problems of BPNN with slow learning and the likelihood of its becoming trapped into a local minimum make it difficult to use in practical applications, especially when the size of the network is large. These problems are due to the fact that the learning process of a standard BP network is mechanical and elementary. It does not have the advanced intelligent characteristics required to generalize the previous training experience.

In this paper, we used two modified BPNN algorithms, Morbidity neurons Rectified BPNN (MRBP) and Learning Phase Evaluation BPNN (LPEPB), to overcome the slow learning and local minima problems which were presented in our previous work (Li & Park, 2009); these two algorithms utilize the concept of learning phase. The whole learning process is divided into many learning phases, with each learning phase containing 50 epochs. Based on experience, the limitations (slow learning and local minima) of the BPNN are related to the morbidity neurons, this is because of the ill-conditioned nature of the standard BP method. The MRBP can detect and rectify the morbidity neurons during each learning phases.

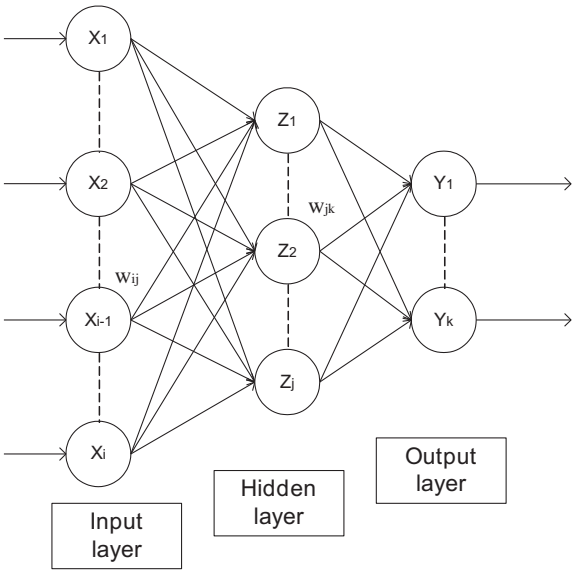

Fig. 1. Typical three layer BP network.