The Improvement of K-NN Classifier with GA-Based Weight-Tunning Method

Wei JIN
State Key Laboratory of Pulsed Power Laser Technology, Electronic Countermeasures Institute, National University of Defense Technology, Hefei, China
kingvee@163.com

ABSTRACT Because the k-Nearest Neighbors (k-NN) algorithm does not take into account different weights values among each feature of data sets, it sometimes achieves low accuracy. In order to increase the k-NN algorithm accuracy, the genetic algorithm-based (GA-based) weight-tunning method is proposed. When the features have been given the same weights values, the k-NN algorithm with EU distance metric is introduced firstly. Secondly, the weighted k-NN algorithm is obtained, when considering different weights values among features. Thirdly, the steps of GA-based weight-tunning method are given in detail. Finally, the experimental tests are running with the real data sets about the students’ knowledge status about the subject of Electrical DC Machines. The experimental results show that the weighted k-NN algorithm achieves classification accuracy of 95.2%, whereas the classification accuracy of k-NN algorithm is 86.2%. The GA-based weight-tunning approach for features of data set could play an essential role in improving the k-NN classifier accuracy.

1. INTRODUCTION
k-Nearest Neighbors (k-NN) algorithm is a non-parametric method used for categorization and regression [1]. The basic k-NN algorithm does not take into account differences among each feature of the inputting data set. It may cause low accuracy in some cases. Because in these cases, features have different effects on the classified labels from each other. Weights values are used to measure differences among the features. Therefore, in order to increase the k-NN algorithm accuracy, each feature can be given a weight value.

In this paper, the training and test data sets which are downloaded from the UCI website[2] are the students’ knowledge status about the subject of Electrical DC Machines. There are five features and four knowledge levels in the data set. Current challenges are to accurately weight the features of students on their knowledge. The approach that the features are weighted by the weight-tunning method which is proposed in reference [3, 4]. The best weight-values may be obtained by genetic algorithm-based (GA-based) method.

The organization of the paper is as follows: Section 2 presents the details of k-NN algorithm. Section 3 presents the principle and steps for creation of GA-based weight-tunning method. The results of experimental study are presented in Section 4. Section 5 concludes the paper.

2. K-NN ALGORITHM
k-NN algorithm works like this: there is an existing set of example data, also called training set. What labels each sample of training set should fall into are known. It is represented by a Matrix, shown in Eq. (1). There are n features of the training set. In Eq. (1), $x_{ij}$ is the jth feature value of the ith sample.
of the training set, and \( l_i \) is the label value of the \( i \)th sample of the training set. A new sample of data with \( n \) features and without a label is given, shown in Eq. (2). \( y \) is a \( n \)-dimensional vector the same as the features number of training set.

\[
T = \begin{bmatrix}
  x_{11} & x_{12} & \cdots & x_{1n} & l_1 \\
  x_{21} & x_{22} & \cdots & x_{2n} & l_2 \\
  \vdots & \vdots & \ddots & \vdots & \vdots \\
  x_{m1} & x_{m2} & \cdots & x_{mn} & l_m
\end{bmatrix}
\]

(1)

\[
y = \langle y_1, y_2, \ldots, y_n \rangle
\]

(2)

The distances among all the samples of training data and the new sample are measured by one mean of distance metric. After that, the sample piece is compared to every sample of training set. The labels of \( k \)-number of neighbors are determined if they are the closest to the new sample. For this purpose, the majority-vote method is commonly used to determine the label of the new sample. The three key parameters of the classification process are the integer ‘\( k \)’, training set and distance metric.

There are three metric distance means which are often used including: Euclidean (EU) distance, Manhattan distance and Minkowski distance. However, Euclidean distance is most popular metric distance of the three means. Therefore, EU distance metric is used for measuring the distance between the feature \( x_i \) and \( y \), shown in Eq. (3).

\[
d(x_i, y) = \sqrt{\sum_{j=1}^{n} (x_{ij} - y_j)^2}
\]

(3)

The \( k \)-NN is used to classify in four steps. The steps are summarized as follows.

i. Determining the \( k \)-value (the number of nearest neighbors for the new sample). \( k \) is an integer and is usually less than 20.

ii. Calculating the distances between the new sample and the training set with Eq. (3), and saving the distance values \( d(x_i, y) \).

iii. Sorting distances \( d(x_i, y) \) and determining of nearest neighbors of the new sample. If \( k = 3 \), the nearest 3 training samples would be obtained.

iv. Making a majority vote among the nearest \( k \) training samples in order to determine the label of the new sample \( y \).

3. GA-BASED WEIGHT-TUNNING METHOD

The basic k-NN algorithm has an assumption that every feature is equally important, and reflects the classification results with the same weights values. However, in some cases, the importance of each feature to reflect its relevance for classification is different. The accuracy of k-NN algorithm would be low if each feature has been given the same weights values in these cases. If features are appropriately weighted with an approach, the performance of the k-NN algorithm will not be degraded. The distance metric that is used for the weighted k-NN is a slight variant of the Euclidean metric (Eq. 3), showed in Eq. 4, where \( w \) are features weights showed in Eq. 5.

\[
d(x_i, y) = \sqrt{\sum_{j=1}^{n} w_j (x_{ij} - y_j)^2}
\]

(4)

\[
w = \langle w_1, w_2, \ldots, w_n \rangle
\]

(5)

The genetic algorithm (GA)-based approach for weight-tuning is used to explore the real-valued weights of features. The genetic algorithm is a metaheuristic inspired by the process of natural selection that belongs to the larger class of evolutionary algorithms. Genetic algorithms are commonly used to generate high-quality solutions to optimization and search problems by relying on bio-inspired operators such as mutation, crossover and selection[5].

Two basic steps of GA-based weight-tuning approach coding of individuals in population and updating population are given in following.

i. Coding of individuals: The weight of each feature can be represented in the form of \( \langle w_1, w_2, \ldots, w_n \rangle \) with a weight vector. The genes of an individual (chromosome) correspond to the
weights and the fitness value of it, showed in Table 1. The last gene ($FV$) is the fitness value of an individual.

Table 1. The genes of chromosome in GA-based weight-tunning method

| Gene_1 | Gene_2 | … | Gene_n | FV |
|--------|--------|---|--------|----|
| $w_1$  | $w_2$  | … | $w_n$  | The fitness of individual |

ii. Creating and updating of population: The flow diagram that shows creating and updating steps of population is given in Fig. 1.

a. Randomly creating population: $h$-item individuals are created randomly to constitute the population. The matrix form of population constituting $h$-item individuals is given in Eq. (6). In Eq. (6), weights values and fitness values ($FV$) are in the interval $[0, 1]$.

$$
P = \begin{bmatrix}
  w_{11} & w_{22} & \cdots & w_{1n} & FV_1 \\
  w_{21} & w_{22} & \cdots & w_{2n} & FV_2 \\
  \vdots  & \vdots & \ddots & \vdots & \vdots \\
  w_{h1} & w_{h2} & \cdots & w_{hn} & FV_h
\end{bmatrix}
$$

b. Calculation of fitness value: It assumes that $num_e$ denotes the number of misclassified samples corresponding to any individual in the set of $P$. Fitness value of this individual is calculated as $FV = 1 / num_e$. Fitness values of all individuals in population are calculated similarly.

c. Termination criteria: In this process, the best individual providing the genetic reproduction finalization criteria among individuals in $P$ population is explored. Termination criteria might be a definite fitness value or a generation number. After the termination of reproduction, the weights of having the best fitness value individual in $P$ will be output as the best weight values of population.

d. Selection of parents: Each individual in population represents a solution for problem. Reproduction operator is used to create new individuals in population. The methods of ‘Roulette Wheel’ will be implemented to select the parents in this paper.

Figure 1. The flow diagram of creating and updating of population.
e. Crossover: It is process that the execution of permutation process vice verse among the genes of children. Thanks to this, the genetic codes of individuals have been changed. In this paper, the crossing methods ‘Flip bit’ is implemented.

f. Mutation: The genetic codes of child individuals are changed. It is used to maintain genetic diversity in $P$ population. In this study, the random-mixed of mutation methods named ‘single point mutation’ are used.

g. Selection of the best individual: It is that the individual of the best fitness value through children is selected.

h. Updating of population: It is that the individual of the lowest fitness value in population is replaced by the best individual.

When the best individual has been obtained, the weights values will be used in Eq. (4) to implement the weighted $k$-NN algorithm.

4. EXPERIMENTS

The training set and testing set are the real data sets about the students' knowledge status about the subject of Electrical DC Machines. The number samples of training set and testing set are 258 and 145 respectively. There are 5 features $<$STG, SCG, PEG, STR, LPR$>$, and what they mean is shown in Table 2. The labels which have 4 categorical values (‘very low’, ‘low’, ‘middle’, and ‘high’) represent the knowledge level of user.

| STG | SCG | STR | LPR | PEG | Labels |
|-----|-----|-----|-----|-----|--------|
| The degree of study time for goal object materials | The degree of repetition number of user for goal object materials | The degree of study time of user for related objects with goal object | The exam performance of user for related objects with goal object | The exam performance of user for goal objects | The knowledge level of user |

The termination criteria of GA-based weight-tunning is set to be the fitness value 0.2 or the generation number 15,000. It means when fitness value of any individual is 0.2 or when the number of generation reaches to 15,000, the weights of individual having the best fitness value will be regarded as the best weights values, and then the updating program of population will be ended. Total number of the created population is 100.

Because the number of features is 5, the number of optimum weights values which may be shown as a vector $(w_{STG}, w_{SCG}, w_{STR}, w_{LPR}, w_{PEG})$ is also 5. The experiments have been performed by the $k$-NN algorithm under the condition that $k$ value is set to 3. The GA-based weight-tunning has searched the 5 optimum weights values for the 5 features of training set. The curves of searching process is presented in Fig. 2. The maximum fitness of individual in $P$ is about 0.14 when the generation number is updated to over 10,000.

![Figure 2. The curve of searching process](image-url)
Table 3. The best weights values for $k = 3$

| $w_{STG}$  | $w_{SCG}$  | $w_{STR}$  | $w_{LPR}$ | $w_{PEG}$ |
|-----------|-----------|-----------|-----------|-----------|
| 3.395e-54 | 2.639e-03 | 7.064e-60 | 0.412     | 0.679     |

The best weight values are in Table 3. When the values are analyzed, it is seen that $W_{PEG}$ and $W_{LPR}$ have larger weights values than the three others. It shows that the two features of students (PEG, LPR) are considerably more efficient than the other three features (STG, SCG, STR) on the knowledge label of users.

One experimental comparison of the knowledge classifiers has been provided. After training on 258 samples, the average number of misclassified samples, the percentage of average error rates and the average classification accuracy of two knowledge classifiers are measured for 145 testing samples. The weighted $k$-NN algorithm achieves an average classification accuracy of 95.2% over the validation set, whereas the classification accuracy of $k$-NN algorithm is 86.2%. Consequently, the weighted $k$-NN algorithm has produced a considerable improvement: the rate of improvement is 9% for $k$-NN classifier.

Table 4. Comparing the performance of two classifiers

|                        | $k$-NN algorithm | Weighted $k$-NN algorithm |
|------------------------|------------------|---------------------------|
| The number of misclassified samples | 7                | 20                        |
| The percentage of error rates      | 4.8%             | 13.8%                     |
| The classification accuracy          | 95.2%            | 86.2%                     |

5. CONCLUSION
This paper presents a powerful and efficient genetic algorithm-based weight-tuning approach to improve the accuracy of $k$-Nearest Neighbors classifier. The experimental studies have shown that when the optimal weights values which are searched by the genetic algorithm-based weight-tuning approach are put into the distance metrics, the classification accuracy of $k$-Nearest Neighbors will be improved 9 pp. It shows that genetic algorithm-based weight-tuning approach for features of data set could play an essential role in improving the $k$-NN classifier accuracy.

ACKNOWLEDGMENTS
My thanks to UCI for allowing me to use their data sets freely.

REFERENCES
[1] http://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm.
[2] http://archive.ics.uci.edu/ml/datasets/User+Knowledge+Modeling.
[3] Kelly J D, Davis L. 1991. A hybrid genetic algorithm for classification. Proceeding of the International Joint Conference on Artificial Intelligence. (1991): 645-650.
[4] H. T. Kahraman, Sagioglu, S., Colak, I. 2013. Developing intuitive knowledge classifier and modeling of users' domain dependent data in web. Knowledge Based Systems. 37 (2013): 283-295.
[5] Bies, Robert R., Muldoon, Matthew F., Pollock, Bruce G., Manuck Steven, Smith Gwennl, Sale, Mark E. 2006. A genetic algorithm-based, hybrid machine learning approach to model selection. Journal of Pharmacokinetics and Pharmacodynamics. (2006): 196–221.