An New *k*-Nearest Neighbors Algorithm Considering Label Uncertainties of Training Samples

Zhengfei Shen and Jian Cao

Shanghai Jiao Tong University, Shanghai, China

Abstract. As a regular non-parametric method used for classification, the k-nearest neighbors algorithm (k-NN) predicts an unlabeled object using the k closest samples whose labels are already known. However, in some cases, the labels given to the samples are not definite. As a result, the accuracy of the predictions can be affected by the uncertainties of the labels of the training samples. In order to improve the performance of k-NN when the labels of training samples are uncertain, we propose a new k-NN algorithm. This algorithm firstly measures the uncertainties of labels. Then, the prediction is made based on the uncertain labels of the k closest samples based on the Dempster-Shafer theory. We also introduce two methods to compute the optimal k value. Experiments on a real-world dataset prove that our algorithm has better performance compared with other algorithms.

Keywords: k-Nearest Neighbors Algorithm · Uncertainty · Dempster-Shafer Theory

1 Introduction

Classification is the problem of identifying to which category or sub-population a new observation belongs, on the basis of a training set of data containing observations or instances whose category membership is known. In many domains, classification is an important task [1][13][14][15]. In this task, the information and features of a sample can be formally and numerically represented as a vector $(x_1, x_2, ..., x_n)$. In such a n-dimensional feature space, the main goal of classification is to estimate the label of a new unlabeled vector by considering the relationships between the labeled and unlabeled vectors.

As a well-known and widely used method, the k-nearest neighbors algorithm (k-NN) method, has been regarded as the most simple but effective method of the many available classification methods such as the Bayesian method, support vector machine and neural networks. Due to its simplicity but good performance, the k-NN has been considered as a top-10 data-mining method [2]. Known as a supervised learning algorithm, its working mechanism is fairly easy to grasp: for a given object, k-NN tries to get the nearest k samples in the training dataset based on a specific distance metric, and then makes predictions according to

the information on these samples. Generally, the object can be classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. Over the past few decades, research has also been carried out to replace the plurality vote method with other approaches, for instance, in [3] [4], an evidence k-NN based on the Dempster-Shafer Theory is proposed to extend the plurality vote method.

Generally speaking, there are two important issues in k-NN. The first relates to the similarity and distance computation between vectors. Recently, many different distance computation algorithms have been proposed. The second issue relates to the selection of the optimal value of k for a given dataset. The general k-NN simply assumes the whole dataset shares a value for k, which is often not proper or inaccurate for a finite and non-uniformly distributed dataset. Recently, many new methods have been proposed to find the value of k in an adaptive and optimal way.

Unfortunately, in all of these efforts, the uncertainty of the training data is neglected. As we know, the training dataset is composed of a group of labeled data samples. However, these labels could be imprecise or even incorrect, despite being provided by experts. Since the label information of the k neighbors will be combined to make the ultimate decision, the label uncertainties of the samples are superimposed so that the accuracy of the k-NN will decrease to some extent. Fortunately, additional and available information can be used to quantify the uncertainty of the label of the sample itself, including the ratings from users or the votes of experts. For instance, Zhu et al. [9] described a real-world decision support system for multi-disciplinary treatment called MdtDSS. The core recommended system predicts a new patient's therapeutic regimen based on the information of other patients. The ultimate therapy of a patient is voted on by a group of doctors which reflects doctors' opinions. The vote from one doctor expresses if he/she approves the ultimate therapy of this patient. In this paper, we define a method to quantify the uncertainties of samples in the training dataset with extra information. Then, we fuse both the uncertainty and the label information as the evidence of this sample in the k nearest neighbors. Then, we use the Dempster-Shafer theory to accumulate all this evidence to decide to which class a test sample belongs. Finally, we propose two strategies to select the optimal value for k for a specific object based on the combined evidence. We perform a series of experiments on a real-world dataset, discussed in [9] to verify our method.

This paper is organized as follows. In section 2, we discuss the related work on k-NN algorithms. In Section 3, we introduce the Dempster-Shafer evidence theory. In Section 4, we present a modified evidence-theory based on the uncertainties of data samples and the two algorithms upon the optimal k value selection for this sample. Finally, in the Section 5, the experiments and results are presented.

2 Related Work

Various studies have been conducted in recent years to improve the performance of k-NN.

Lin et al. [5] proposed a similarity computing method via fusing neighborhood information. In this method, both the Euclidean distance and the neighbor information are considered simultaneously, and the two metrics are combined together to measure the similarity between samples. Yung-Kyun Noh et al. [6] proposed a generative metric learning method to enhance the performance of the k-NN algorithm. To select the optimal value for k on a given dataset, Nicols Garca-Pedrajas et al. [7] proposed an algorithm to obtain local k value with a simple and fast procedure. Zhang et al. [8] proposed to learn a correlation matrix to reconstruct test data points by training data to assign different k values to different test data points.Min-Ling Zhang [16] presented a multi-label lazy learning approach named ML-KNN.R. Li [17] presented a new algorithm that combined Support Vector Machine with K Nearest neighbour.Cui Yu [18] proposed an efficient method, called iDistance, for K-nearest neighbor (KNN) search in a high-dimensional space.Shan-Hung Wu [19] study the multidimensional similarity query for large-scale sensor networks and propose a new algorithm called DIC (dimension reduction by Chebyshev polynomials).

The plurality vote method is the simplest rule to combine the labels of k nearest neighbors and is based on the assumption that each of the k nearest neighbors are equally important. In practice, the circumstances can be more complex. Intuitively, the closer the neighbor, the more possible that the unknown object will be in the class of this neighbor. In 1995, Denux [3] defined the frame of discernment [10] of a k-NN method and used the distance of samples to measure the mass function of one sample. Our work is an extension of his work. To replace the plurality vote rule, some researchers investigate relationships between the global and local probability distributions. For a small training dataset, Cheamanunkul [11] proposed a simple k-NN rule that takes into account the labels of all of the neighbors, rather than just the most common label. In his approach, relative entropy is used to measure the relationship between the global and local probability distribution.

Unfortunately, the previous studies neglect the uncertainties of the labels of training samples, which is the topic of this paper.

3 Preliminary

3.1 Dempster-Shafer theory

As a generalization of the Bayesian theory of subjective probability, the Dempster-Shafer theory was proposed in 1976 in [10]. It is a general framework for reasoning with uncertain data. This theory provides a model to combine evidence from different sources to take into account all the available evidence. It comprises two crucial steps: a basic belief assignment(BBA) and evidence combination.

In the first step, let U represent a non-empty set of mutually exclusive and exhaustive propositions called the frame of decrement. The power set 2^U is all the subsets of the set U, which includes both the empty set and the entire set U. For the frame of discernment U, the function $m{:}2^U \rightarrow [0,1]$ is a basic probability assignment (BPA), also called a mass function. This function satisfies the following two conditions:

$$\begin{aligned} &(1)\ m(\phi) = 0 \\ &(2)\ \sum_{A \subset U} m(A) = 1 \end{aligned} \tag{1}$$

$m(A)$ indicates the exact degree of trust in A. The subsets A of U where $m(A) > 0$ are called the focal elements of the belief function, and their union is called its core. The function $Bel{:}2^u \rightarrow [0,1]$ is the belief function over U and is defined as the sum of all the masses of subsets:

$$Bel(A) = \sum_{B \subset A} m(B) \tag{2}$$

Belief (usually denoted Bel) measures the strength of the evidence in favor of a proposition but not any of its subsets. It ranges from 0 (indicating no evidence) to 1 (denoting certainty). An important difference with probability theory is the sum of belief of a proposition and its negation not necessarily equals to 1. Hence, the remaining belief of A, called total ignorance, is the belief of the whole frame of discernment.

In the second step, the Dempster-Shafer theory defines a method to combine two or more mass assignments of the same frame of discernment in specific situations. To combine the assignments means to accumulate all the evidence from all sources in one frame. This rule derives the common shared beliefs among multiple sources and ignores all the conflicting (non-shared) beliefs through a normalization factor.

Given n mass functions m_1,m_2...m_n on the same frame of discernment U, for arbitrary A included in U, the combination rule is as follows:

$$\begin{aligned} &m_{1,2,\ldots,n}(\phi) = 0 \\ &m_{1,2,\ldots,n}(A) \ = (m_1 \oplus m_2 \oplus \cdots \oplus m_n)(A) \\ &\qquad\qquad = \frac{1}{K} \sum_{A_1 \cap A_2 \cap \cdots \cap A_n} m_1(A_1) \cdot m_2(A_2) \cdots m_n(A_n) \end{aligned} \tag{3}$$

where:

$$\begin{aligned} K \ &= \sum_{A_1 \cap A_2 \cap \cdots \cap A_n \neq \phi} m_1(A_1) \cdot m_2(A_2) \cdots m_n(A_n) \\ &= 1 - \sum_{A_1 \cap A_2 \cap \cdots \cap A_n = \phi} m_1(A_1) \cdot m_2(A_2) \cdots m_n(A_n) \end{aligned} \tag{4}$$

the K is called a normalization factor which is a measure of the number of conflicts among the mass functions.

3.2 Dempster-Shafer Theory based *k*-NN

Also known as the evidence-theory based k-NN (EKNN) , the Dempster-Shafer (DS) theory based k-NN was proposed by Thierry Denux [3] in 1995. He first established the connection between the multidimensional vector space in the k-NN algorithm and the frame of discernment in DS evidence theory. In this approach, each neighbor of a pattern is considered as evidence supporting some hypotheses about the class membership of that pattern. The BPAs are calculated for each of the k nearest neighbors. The belief of each hypothesis is obtained by aggregating BPAs using Dempsters rule of combination. The contributions can be summarized into two points: (1) a generalized way to compute the BPA value of each k nearest neighbor respectively based on the distance to the unlabeled data samples; (2) the application of the combination rule into the k-NN with a special form. Our work is also based on the DS evidence theory and the considerations of uncertainties of neighbors are an extension to Thierry's work which is presented in this section.

In a classification problem, the training dataset can be regarded as a collection of N p-dimensional training samples represented by $X = \{x^i = (x_1^i, ..., x_p^i)|i = 1, ..., N\}$ and every sample in the dataset belongs to one and only one class from M classes $C = \{C_1, ..., C_M\}$. In the very beginning, each sample in the training data is labeled a class in C with a certain degree of uncertainty, which is discussed in detail in the following section. The labeled dataset can be represented as a binary relationship (X, L), where L is the set of labels, which can be used to classify new patterns.

In the Dempster-Shafer theory based k-NN, all possible class sets C make up the frame of discernment, to predict the true label of an unlabeled pattern x^s.

For this unlabeled new pattern x^s, the k neighbors based on a specific distance measurement make up a set Φ^s. Each neighbor in Φ^s provides a piece of evidence as to whether x^s belongs to a specific class Cq in C, The negation of this piece of evidence is totally innocent, i.e. it doesn't refer to C itself rather than any subsets of C. If we use the mass function to represent this piece of evidence, we obtain the following relationship:

$$\begin{aligned} m^{s,i}(\{C_q\}) &= \alpha_q \\ m^{s,i}(C) &= 1 - \alpha_q \end{aligned} \tag{5}$$

where $i = 1, 2, ..., k$.

Definition 1. *α_q reflects how much the intensity is that neighbor x_i supports the classification of unlabeled pattern x^s as C_q.*

According to definition 1, α_q should be a function of the distance from x_i to x_s, because the smaller the distance between x_i and x_s, the more crucial it is to decide whether the class of x_s is the same as the class of x_i. Moreover, as the distance between x_s and x_i becomes infinitely large, the belief function given by $m^{s,i}$ makes no sense, which means that ones belief concerning the class of x_s is no longer affected by ones knowledge of the class of x_i.

We can replace α_q with any reasonable decreasing function t. Here we use the following one:

$$\alpha_q(d^{s,i}) = \alpha_0 e^{-d^{s,i}\beta} \tag{6}$$

Then, we can get all the k mass functions of the k nearest neighbors respectively so that the distance to x_s is available. To make the ultimate decision as to which class x_s belongs, we first combine all these pieces of evidence together, based on the DS evidence combination rule. According to equation (3), for each class q, we get:

$$\begin{aligned} m_q^s(\{C_q\}) &= 1 - \prod_{x^i \in \Phi_q^s} (1 - \alpha_q(d^{s,i})) \\ m_q^s(C) &= \prod_{x^i \in \Phi_q^s} (1 - \alpha_q(d^{s,i})) \end{aligned} \tag{7}$$

Note that here the BPA function $m_q^s(\{C_q\})$ measures the combined belief according to all the neighbors whose class is C_q.

Combining all the BPAs m_q^s for each class, a global BPA $m^s = \oplus_{q=1}^M m_q^s$ is obtained as:

$$\begin{aligned} m^s(\{C_q\}) &= \frac{m_q^s(\{C_q\}) \prod_{r \neq q} m_r^s(C)}{K} \\ m^s(C) &= \frac{\prod_{q=1}^M m_q^s(C)}{K} \end{aligned} \tag{8}$$

where K is the normalizing factor $q = 1, 2, .., M$.

In the last step of the evidence rule-based k-NN, for each C_q in C we compute its BPA according to equation (8), and then assign x_s with the optimal class.

4 Uncertainty and Evidence Theory-based *k*-NN Algorithm (UCEkNN)

In this section, we present our own k-NN algorithm based on uncertainty and evidence theory (UCEkNN), and then we propose two optimal k value selection algorithms, based on the UCEkNN.

4.1 UCEkNN

According to equation (5), up untill now, we used α_q to represent the BPA of one neighbor x^i among k nearest neighbors set whose class is C_q. In practice, however, there is some uncertainty of the label itself. While the BPA of each neighbor is combined, the uncertainties are also accumulated, and the accuracy of the ultimate prediction decreases to some extent. From this perspective, the uncertainty of the label cannot be ignored and we denote the uncertainty as UC:

Definition 2. *UC^i whose value ranges from 0 to 1 describes the uncertainty of the label itself of the neighbor x^i among the k nearest neighbors of x^s. UC^i depends on the neighbors characteristics only.*

According to definition 2 and equation 5, we propose a new form of the BPA function:

$$\begin{aligned} m^{s,i}(\{C_q\}) &= \alpha_q \cdot UC^i \\ m^{s,i}(C) &= 1 - \alpha_q \cdot UC^i \end{aligned} \tag{9}$$

Similarly, according to the DS combination rule, for each class q:

$$\begin{aligned} m_q^s(\{C_q\}) &= 1 - \prod_{x^i \in \Phi_q^s} (1 - \alpha_q(d^{s,i}) \cdot UC^i) \\ m_q^s(C) &= \prod_{x^i \in \Phi_q^s} (1 - \alpha_q(d^{s,i}) \cdot UC^i) \end{aligned} \tag{10}$$

Although UC^i was discussed in [3] as imperfect labeling, our approach is intrinsically different. In our approach, we do not change the basic form of equation (5), i.e. we assume one neighbor only belongs to one specific class in spite of uncertainty rather than two or more possible classes. Moreover, we asuume UC^i is independent of the distance and can be accessed easily with some extra information of x^i. Here, we use an opinion set to describe the extra information, denoted as OpS:

Definition 3. *OpS^i describes the extra information set of x^i, which can be a set of votes information, ratings information, or other reasonable information which illustrates the experts' or users' subjective opinions on whether x^i should be labeled with C_q.*

In practical applications, for the k-NN, the label of each sample is decided by a group of people, usually experts. OpS^i is such a set which contains the opinions. OpS is accessible in many problems. For example, [9] introduces a medical therapy recommendation system based on the k-NN algorithm. For each patient in the training dataset, the ultimate therapy is voted on by at least five doctors. The therapy most voted for is used to label this patient. Here, the doctors' vote results make up the OpS of this patient.

Intuitively, the consistency of doctors' opinions reflects the complexity of making decisions. If all doctors vote for the same therapy, this patient's case is not complex and it is easy to make a decision, however in situations where there are different votes, this patient's case is complex and controversial.

We use the information entropy(IE) [12] of OpS^i to quantify its consistency:

$$H(OpS^i) = -\sum_{c \in C} P_{ic} \cdot log(P_{ic}) \tag{11}$$

where P_ic is the proportion of class c in OpS^i.

Moreover, the uncertainty UC^i of the label of x_i has some connections with the uncertainty of OpS^i. For a highly consistent case where $H(OpS^i)$ has a small value, the UC^i should be closer to 1 while for a controversial case where $H(OpS^i)$ has a large value, the UC^i should be closer to 0. From this perspective, we conclude the UC^i is a decreasing function of $H(OpS^i)$. We suggest chosing the following function:

$$UC^i = UC_0 e^{-H(OpS^i)\beta_u} \tag{12}$$

For a query instance of unknown category x^s, the prediction process is shown in Algorithm 1.

Algorithm 1 Outline of the proposed Algorithm

Input: a training set $T = \{(x_1, y_1), ..., (x_n, y_n)\}, x_i \in \Re^p$, a reasonable k, a query instance x^s, OpS^i for each sample in training set
Output: the optimal label of x^s

```
get the k nearest neighbors set N = {(x_1, y_1), ..., (x_k, y_k)}, x_i ∈ ℜ^p
for all x^i in T do
    calculate H(OpS^i) for each x^i using equation (11)
    calculate BPA for each x^i using equation (9)
end for
for all classification C_q in C do
    calculate combining BPA ms({C_q}) of C_q
end for
return the optimal label whose BPA ls the largest
```

4.2 Optimizing Values of *k* for UCEkNN

Traditional value selection algorithms for k aim to find the global optimal k for a fixed training dataset and testing dataset. Although a good value might be obtained using cross-validation (CV), the same value is unlikely to be optimal for the whole space spanned by the training set. In this work, we devise a new greedy method for UCEkNN.

For different k ranging in $[k_min,k_max]$, the neighbor set of x^s is different, as is the BPA of different classifications. Consequently, searching for an optimal k for an unlabeled object is tantamount to determine the best opportunity when the BPA of all classifications meet some optimal conditions. Here, we provide two available conditions:

Condition 1 For different k, the largest value of $ms(\{C_q\})$ reaches the maximum.

Condition 2 For different k, the difference between the largest and second largest value of $ms(\{C_q\})$ reaches the maximum.

The execution steps of our approach under condition 1 and condition 2 are detailed in Algorithm 2 and Algorithm 3 respectively.

Algorithm 2 Algorithm for Condition 1

Input: a training set $T = \{(x_1, y_1), ..., (x_n, y_n)\}, x_i \in \Re^p$, a reasonable k, a query instance x^s, OpS^i for each sample in the training set
Output: the optimal label of x^s

for all k in [k_{min},k_{max}] **do**
  for all classify x^i in T **do**
    calculate $H(OpS^i)$ for each x^i using equation (11)
    calculate BPA for each x^i using equation (9)
  end for
  for all classify C_q in C **do**
    calculate combining BPA ms($\{C_q\}$) of C_q
  end for
  get $m^s(A) = max\{m^s(\{C_q\}), C_q \in C\}$
end for
return the optimal label whose BPA is the largest

Algorithm 3 Algorithm for Condition 2

Input: a training set $T = \{(x_1, y_1), ..., (x_n, y_n)\}, x_i \in \Re^p$, a reasonable k, a query instance x^s, OpS^i for each sample in training set
Output: the optimal label of x^s

for all k in [k_{min},k_{max}] **do**
  for all classify x^i in T **do**
    calculate H(OpS^i) for each x^i using equation (11)
    calculate BPA for each x^i using equation (9)
  end for
  for all classify C_q in C **do**
    calculate combining BPA $ms(\{C_q\})$ of C_q
  end for
  get $m^s(A) = max\{m^s(\{C_q\}), C_q \in C\}$
  get $m^s(B) = max\{m^s(\{C_q\}), C_q \in C \; and \; C_q \neq A\}$
end for
return the optimal label whose BPA is the largest

4.3 Optimization of UCEkNN Using *L*-Sure Algorithm

The influence of different experts' and users' opinions on the data sample's label should not be identical. For instance, in relation to a medical decision, the opinion

of an experienced and proficient doctor is more crucial. From this perspective, simply using the entropy of OpS to measure uncertainty is no longer reasonable.

The L-Sure algorithm is devised to intensify the opinions from more authoritative people who vote for the labels of samples. The basic idea of the L-Sure algorithm can be described as the following two steps: (1) to select the L-most authoritative experts of the vote set; and (2) to calculate uncertainty based on the opinions of these L experts. Given an unlabeled sample and its k neighbors, a straightforward strategy to pick the L-most reliable experts is to sort all the experts by the precision in the k neighbors from high to low. Such an idea is easy and efficient.

After obtaining the L experts for an unlabeled sample, uncertainty can be calculated by the following conditions: (1) if all the L experts hold the same opinion, then the UC_i in equation (9) equals 1; (2) if one or more experts out of L experts hold different opinions, then we still use equation (12) to calculate the UC^i.

All these aforementioned steps are described in equation (13):

$$UC^i = \begin{cases} UC_0 & \text{condition 1} \\ UC_0 e^{-H(OpS^i)\beta_u} & \text{condition 2} \end{cases} \tag{13}$$

5 Experiments and Discussions

5.1 Experiments for UCEkNN and Optimal k Values

To test our proposed two models (the original UCEkNN and UCEkNN with optimal k values), we conduct two experiments which is **Experiment I** (for UCEkNN) and **Experiment II**(for UCEkNN with optimal k values) using the same real-world dataset as discussed in the MdtDSS mentioned in [9].The dataset consists of patients with their clinical properties as the feature vectors and the treatment as the label. The size of the dataset is 3340 if we only use it to examine the general kNN, however, only 1455 of them contain voting information which can be used in our method.

In the 1455 instances which have OpS, the average size of OpS is 8 which means there are 8 doctors voting to one patient on average. We repeat the two experiments three times respectively based on three different subsets of the 1455 instances. Each of the subsets are divided into three segments which have different functions. The three segments are the training dataset, the verification dataset and testing testing data set. **Experiment I and II** use different parts of the three segments: (1)**Experiment I** only uses the training dataset; (2) **Experiment II** uses the training dataset, verification dataset and testing dataset at the same time. The length of each segment during the three experiments is illustrated in Table 1.

Figure 1 reflects the distribution of information entropy in the data set. All sample points are involved in statistics. The figure indicates that the number of sample points with an information entropy of 0 is the largest. That is to say, the

Table 1. Length of each segment during the three experiments.

experiment	training	verification	testing
1	1028	212	215
2	790	238	427
3	610	347	239

labels of most samples are not uncertain. But there are many samples that are very uncertain.

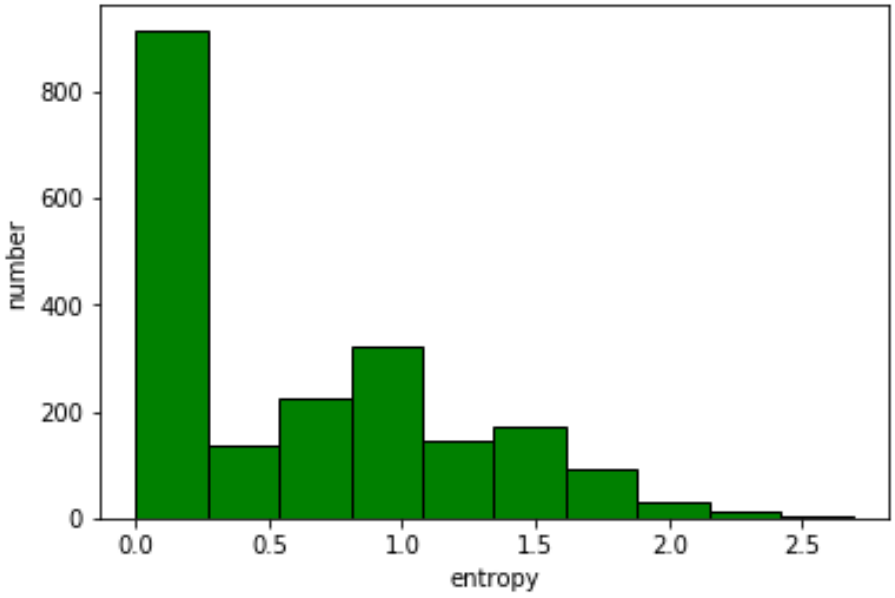

Fig. 1. The distribution of information entropy in the data set. All sample points are involved in statistics.

In **Experiment I**, we predict the label of each instance in the testing dataset using two methods: (1) the basic k-NN method (kNN)(2) the ordinary DS evidence rule-based k-NN (EkNN), and (3) our proposed uncertainty-based k-NN (UCEkNN). In this experiment, we use the globally optimal k as the input of these two algorithms. We devise a simple greedy algorithm to select k in $[k_{min},k_{max}]$. We compare the predicted result and the actual result to judge if the prediction is correct and then calculate the accuracy of this experiment. The results of these three experiments are shown in Table 2.

In **Experiment II**, we compare the performance of UCEkNN with different k and with a fixed globally optimal k. Thus, we use (1) the fixed globally optimal k,

Table 2. Comparison Results of kNN EkNN and UCEkNN

Experiment	kNN	EkNN	UCEkNN
1	54.32%	59.53%	66.05%
2	58.91%	63.00%	66.04%
3	55.66%	58.16%	64.01%

(2) different k acquired by Algorithm 2, or (3) different k acquired by Algorithm 3 to predict the label of each instance in the testing dataset based on UCEkNN and compute the accuracy. The results are shown in Table 3.

Table 3. Comparative Results of UcEkNN with Different Strategies to Select k

experiment	fixed k	diff k based on Alg.1	diff k based on Alg.2
1	66.05%	70.70%	69.30%
2	66.04%	69.79%	69.09%
3	64.01%	67.36%	67.78%

Figure 2 shows the process we decide which is the optimal k value when we predict with the basic k-NN method. We conduct 3 experiments on above 3 data set using successive k from a fixed interval of k and plot the precision using that k. When the precision get its maximum, we get the optimal k value of that data set.

In the **Experiment II**, an unlabeled object gets its label and then becomes a historical instance and turns into part of the training set. It is inflexible and inadvisable to keep a fixed k or calculate a new optimal k at a regular interval. In contrast, our proposed method achieves better performance in terms of accuracy.

5.2 Experiments using the L-Sure Algorithm

To verify the efficiency of the L-Sure algorithm and to intensify the comparison with the normal UCEkNN algorithm, we conduct an experiment using the same dataset as that used in experiment 2, and the capacity of the training, verification, and testing dataset is 610,347 and 239 respectively. In our experiment, the value of L varies in the range of 0 to 16, and the result is shown in Fig.3:

6 Conclusions

Based on the Dempster-Shafer evidence theory, we proposed our own model which considers the uncertainty of the label of each sample as a significant factor undertaking classification. Later. We extended our algorithm to optimize the

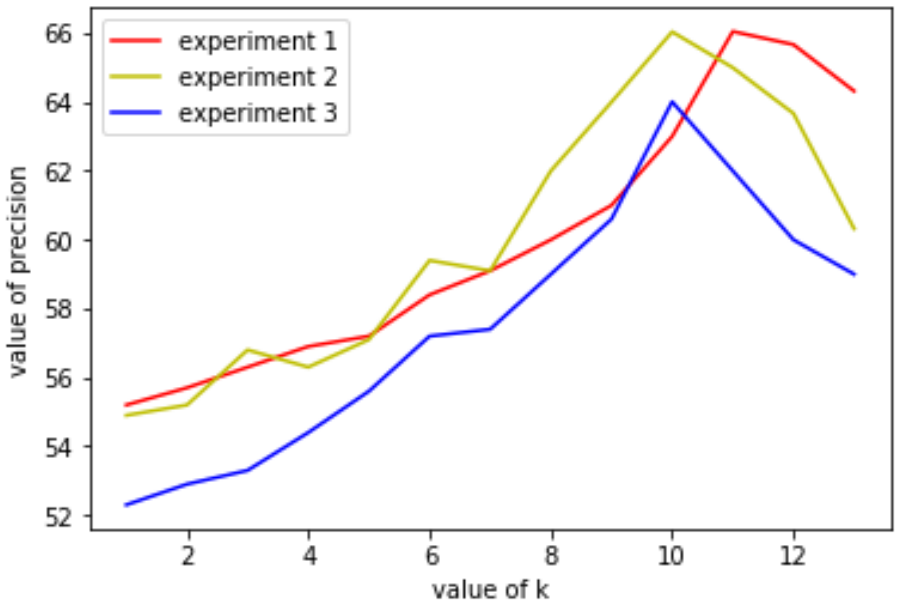

Fig. 2. The process we decide which is the optimal k value when we predict with the basic k-NN method.

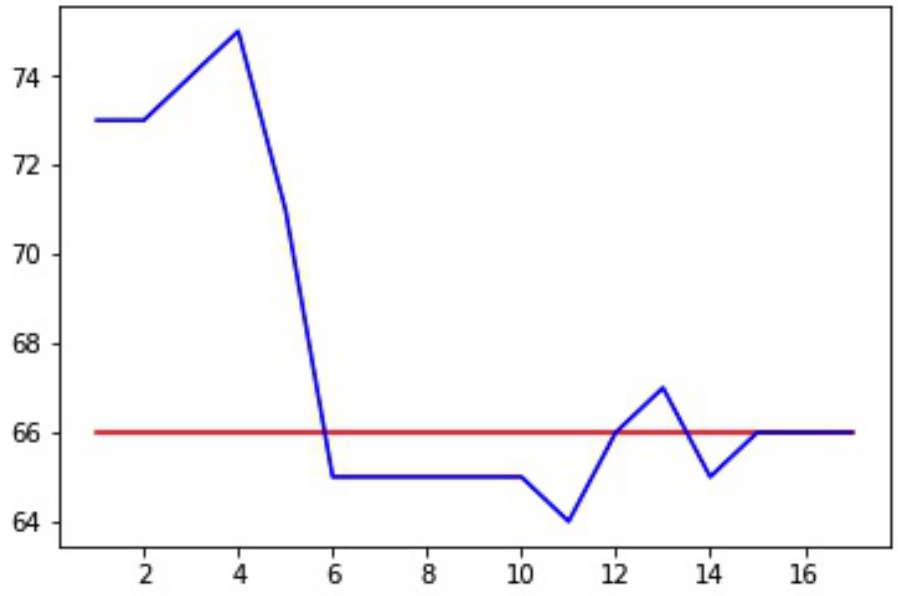

Fig. 3. The red curve represents the normal UCEkNN and the blue curve represents UCEkNN with the L-Sure algorithm. This figure shows that when L meets 6 to 11, the error time is lower than that the error time without using the L-Sure algorithm.

k in the kNN algorithm and devised the L-sure algorithm to intensify the opinions from more authoritative people who vote for the labels of a samples. The experiments show that our algorithm outperforms the regular kNN algorithm with the DS rule.

References

1. Zhu, Xiaofeng , X. Li , and S. Zhang . "Block-Row Sparse Multiview Multilabel Learning for Image Classification." IEEE TRANSACTIONS ON CYBERNETICS 46.2(2016):450.
2. Wu, Xindong , et al. "Top 10 algorithms in data mining." Knowledge and Information Systems 14.1(2008):1-37.
3. Denoeux, Thierry . "A k-nearest neighbor classification rule based on Dempster-Shafer theory." Systems Man & Cybernetics IEEE Transactions on 25.5(1995):804-813.
4. Wang, Lei , L. Khan , and B. Thuraisingham . "An Effective Evidence Theory Based K-Nearest Neighbor (KNN) Classification." 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology IEEE, 2009.
5. Lin, Yaojin , et al. "A new nearest neighbor classifier via fusing neighborhood information." Neurocomputing 143(2014):164-169.
6. Noh, Yung Kyun , B. T. Zhang , and D. D. Lee . "Generative Local Metric Learning for Nearest Neighbor Classification." IEEE Trans Pattern Anal Mach Intell PP.99(2018):106-118.
7. Garciapedrajas, Nicolas , J. A. R. Del Castillo , and G. Cerruelagarcia . "A Proposal for Local k Values for k-Nearest Neighbor Rule." IEEE Transactions on Neural Networks & Learning Systems 28.2(2015):470.
8. Zhang, Shichao , et al. "Learning k for kNN Classification." ACM Transactions on Intelligent Systems and Technology 8.3(2017):1-19.
9. Zhang, Yan , et al. "A Multi-disciplinary Medical Treatment Decision Support System with intelligent treatment recommendation." IEEE International Conference on Computer & Communications IEEE, 2017.
10. Rota, Gian Carlo . "297 ppG. Shafer, A Mathematical Theory of Evidence, Princeton University Press (1976). " Ade Bulletin 2(1977):N/A.
11. Cheamanunkul, S , and Y. Freund . "Improved kNN Rule for Small Training Sets." International Conference on Machine Learning & Applications IEEE, 2014.
12. R Grey. Entropy and Information Theory. ENTROPY AND INFORMATION THEORY. 2012.
13. Chunxia W . CLASSFICATION AND IDENTIFICATION OF THE PLANT GROWTHPROMOTING RHIZOBACTERIA (PGPR) IN COTTON[J]. JOURNAL HUAZHONG(CENTRAL CHINA) AGRICULTURAL UNIVERSITY, 1997.
14. Matarasso A . Abdominolipoplasty: A system of classfication and treatment for combined abdominoplasty and suction-assisted lipectomy[J]. Aesthetic Plastic Surgery, 1991, 15(1):111-121.
15. Khan S M, Islam R, Chowdhury M U. Medical image classfication using an efficient data mining technique[C]// International Conference on Machine Learning Applications-icmla. 2004.
16. Zhang M L , Zhou Z H . ML-KNN: A lazy learning approach to multi-label learning[J]. Pattern Recognition, 2007, 40(7):2038-2048.

17. Li R , Ye S W , Shi Z Z . SVM-KNN classifier - A new method of improving the accuracy of SVM classifier[J]. Acta Electronica Sinica, 2002, 30(5):745-748.
18. Yu C , Ooi B C , Tan K , et al. Indexing the Distance: An Efficient Method to KNN Processing[J]. Vldb, 2001.
19. Wu S H , Chuang K T , Chen C M , et al. DIKNN: An Itinerary-based KNN Query Processing Algorithm for Mobile Sensor Networks[C]// Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on. IEEE, 2007.