IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

Survey on KNN and Its Variants

Alka Lamba[1], Dharmender Kumar[2]

Student, Department of Computer Science and Engineering, GJU S&T, Hisar, India [1]

Associate Professor, Department of Computer Science and Engineering, GJU S&T, Hisar, India [2]

Abstract: This paper discusses about KNN algorithm and its various modified versions available in formerly done studies. Despite the simplicity of KNN algorithm there are a number of shortcomings in it i.e. high computational cost, large memory requirement, equal-weighted features and in last deciding appropriate value of the input parameter k. Many researchers have proposed modified versions of KNN algorithm to overcome these shortcomings. In this study, we will analyze profoundly these variants of KNN algorithm and their performances.

Keywords: Evolutionary Computing; KNN; lazy learner; metaheuristic; nearest neighbour.

I. INTRODUCTION

The necessity of data mining techniques has emerged quite immensely nowadays due to massive increase in data. Data mining is the process of extracting patterns and mining knowledge from data [1]. The data mining is part of KDD process [2]. By KDD we mean Knowledge Discovery from Data. KDD is discovery of new automated, comprehensible patterns which have futuristic value. The applicability of data mining can be seen in many fields such as finance, sales, medicine, marketing, healthcare and insurance, banking [3],[4],[5]. Mining of knowledge from data can be carried out using data mining techniques such as classification, clustering, prediction, association rule, neural networks. Each data mining technique can be practised in specific areas. Classification is used for fraud detection, bank loan categorization etc. Clustering is used to group similar kind of objects. Prediction is used for predicting continuous valued class attribute. Association rule is used basically to find correlation between objects for example in market-basket analysis. Neural network is used in case of handwriting recognition, pattern recognition etc. [6] discusses about these data mining techniques and their applications. Classification is a data mining technique which is used to build a model that can predict future trends. Classification is an example of supervised learning. It is so because the data set given as input is class labelled. There are two steps involved in the process of classification. The first step is to construct a classifier using training data set. This step is called learning step. The second step is the assessment of the built classifier. For executing the process of classification there are a number of algorithms such as decision tree induction, Bayesian classifiers, k-nearest neighbour algorithm, classification using back-propagation, rule-based classifiers, support vector machines (SVM) [7]. [8] projects a comparative study of these classification techniques. Nearest Neighbour is the oldest and very simple technique used for classification.

According to the nearest neighbour technique the classification of an unknown data tuple is accomplished by analysing the classes of its nearest neighbours. KNN algorithm employs this principle of nearest neighbour technique. But in case of KNN algorithm a fixed number of nearest neighbours are allowed to vote in the process of classification of an unknown data tuple which is identified by k, where k is a positive integer. When k=1 then the unknown data tuple is classified as the class of the training data tuple which is most nearest to it. KNN is a non-parametric lazy learner [9]. Opposite to the parametric methods the shape of the classification model is not presumed in case of non-parametric methods. KNN is called lazy learning algorithm while rest of classification algorithms are called eager ones because it does not construct a classifier after getting training data tuples as normally done in first step of classification. Precisely speaking, there is no explicit training phase in KNN. It starts working only when it gets an unseen tuple for classification. Due to this working principle of KNN instance-based learner is another name of it.

k-Nearest Neighbour algorithm is applicable in a number of fields such as pattern recognition, text categorization, finance, agriculture, medicine etc. [10] discusses about how KNN can be used for prediction of economic events. KNN has been used for diagnosing heart disease [11]. [12] demonstrates how well KNN algorithm works in case of intrusion detection. KNN algorithm can be used for both classification and regression. In case of classification the class attribute is of categorical type while in regression it is a continuous variable. [13] has implemented KNN for regression to predict the prices of stock. KNN algorithm has been used for text categorization [14]. It used standard KNN algorithm for classification of documents. KNN algorithm has also been used for ranking documents. [15] has employed KNN for query dependent ranking of documents.

The favourable points associated with KNN are its simplicity, comprehensibility, scalability. It gives good accuracy which is comparable to the results given by the other methods used for classification. Also its robustness towards noisy data makes it a strong contender. But besides many favourable points it has some adversities also. The KNN algorithm needs one input parameter which is k. This parameter plays an important role to grab

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

good accuracy. But deciding the appropriate value of k is very tricky because the smaller value of k would result in high variance and large value of k would result in high bias [16]. And we know that to construct a good model we have to achieve equilibrium between variance and bias. The other thing is KNN's lazy behaviour. It requires enough amount of memory so that all the training tuples can be stored. It becomes very problematic in case of large sized data sets. And last but not the least is its large cost of computation. The cost of computation rises because each time when an unknown tuple is to be classified we have to calculate its distance from all the training tuples in the data set to get k-nearest training tuples to it. So, the computation in KNN is well suited to be performed on parallel hardware. This would save time also. In this research paper we are going to study various other modified nearest neighbour algorithms, proposed in previously done studies, which tried to alleviate infirmities of the standard KNN algorithm

II. K-NEAREST NEIGHBOURS ALGORITHM (KNN)

Out of the various algorithms used for classification, the KNN is the most commonly used algorithm. KNN algorithm is very easy to implement and give fairly good results. Also KNN algorithm does not require any prior knowledge regarding data set for classification. It performs classification purely on similarity basis. As the name of algorithm indicates, for classification of a novel tuple it looks out for k nearest tuples to it. The procedure of classification in KNN starts with a data set. The data set is constituted of certain number of attributes that define a data set. The data set is divided into two sets: training set and test set. Training set is given as input to the algorithm while test set is used to calculate the accuracy of algorithm. The division of the data set can be done using various methods such as hold-out method, random sampling, cross validation etc. KNN classifies any new tuple by using training data tuples similar to it. Due to this KNN is also called local learner. There is no learning phase in KNN. It stores all the training tuples given to it as input without doing anything. All the computations are done at the time of classification of a test tuple. In KNN algorithm, the training tuples can be viewed as a set of data points in an n-dimensional space, where n dimensions are the set of n attributes describing the data set. When an unknown tuple comes for classification, we have to find out the k most nearest data points to it in the n-dimensional space. To find the k most nearest data points to the unknown tuple various distance metrics are used for example Euclidean distance, Minkowski distance, Manhattan distance. The Euclidean distance between two data tuples X and Y is given below:

$$\sqrt{\sum_{1\leq i\leq n}(x_i - y_i)^2} \qquad (1)$$

Manhattan distance between data tuples X and Y is computed as:

$$\sum_{1\leq i\leq n}|x_i - y_i| \qquad (2)$$

Minkowski distance between data tuples X and Y is computed as:

$$\left(\sum_{1\leq i\leq n}|x_i - y_i|^p\right)^{1/p} \qquad (3)$$

where p is a positive integer.

The pseudo code for the KNN algorithm is given below:

ALGORITHM I

Input Parameters: Data set, k

Output: Classified test tuples

Step 1: Store all the training tuples.

Step 2: For each unseen tuple which is to be classified

A. Compute distance of it with all the training tuples using equation no. (1).

B. Find the k nearest training tuples to the unseen tuple.

C. Assign the class which is most common in the k nearest training tuples to the unseen tuple.

End for

Before applying formula (1) to compute Euclidean distance, we have to normalize values of all the attributes of the data set. This would result in easy calculations. The normalization of all the values of an attribute is done in range [0, 1]. 1 would represent the highest value of the attribute and 0 representing the lowest one. Generally we use min-max normalization which is computed as:

$$v' = (v - min_A)/(max_A - min_A) \qquad (4)$$

Here value v of an attribute A is normalized in a new range [0, 1]. max_A, min_A are maximum and minimum values of the attribute A respectively.

From the algorithm we can see that KNN would need two input parameters i.e. data set and k. The input parameter k will decide that how many nearest neighbours are to be undertaken in the process of classification. The class of the unseen tuple would be the majority class in the k-nearest neighbours to the unknown tuple. If the class attribute of the data set is real–valued then the average of the values of class attribute of k nearest training tuples would be assigned to the class attribute of unseen tuple. The prediction of real-valued class attribute is called regression. The simplest form of KNN is the nearest neighbour rule where the value of the input parameter k is taken as 1. In that case, the class of the most nearest training data point to unknown tuple is the class of the unknown tuple too. So in last we can conclude that there are number of factors that would be affecting the performance of KNN algorithm i.e. what should be the value of k, which distance metric should be used, should all the attributes be equal weighted.

III. VARIANTS OF KNN

As discussed previously, there are many shortcomings associated with KNN algorithm. By making changes to the influencing factors the performance of KNN can be enhanced. There are many variants of KNN algorithm proposed in previously done studies which tried to overcome these shortcomings. Some of them are described in the subsequent part of this section.

A. Locally Adaptive KNN

[17] proposed an algorithm called Locally Adaptive k-nearest neighbour algorithm. In standard KNN algorithm,

 DOI 10.17148/IJARCCE.2016.55101

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

global value of input parameter k is used. But this proposed algorithm suggested using different values of the parameter k for different portions of input space. Each time for classification of a query, the value of k is determined via applying cross-validation in its local neighbourhood.

B. Weight Adjusted KNN
In standard KNN algorithm all the attributes have equal importance. All the attributes or features give equal contribution for classification of novel tuples. But not all the attributes in the data set are equally important. Therefore [18] has proposed a weight-adjusted KNN algorithm which first learns weights for different attributes and according to the weights assigned, each attribute would affect the process of classification that much only.

C. Improved KNN for Text Categorization
As we know how much the value of input parameter k influences the performance of KNN algorithm. So, it is very crucial to choose appropriate value of the parameter k. In general the classes are not evenly distributed in the data set. Therefore, using a fixed value of k for all the classes would result in bias towards the class which has larger number of tuples. [19] has suggested to use different values of k for different classes according to their class distribution. More number of tuples is used to classify a new tuple in a class which has large number of tuples.

D. Adaptive KNN
[20] proposed rather than using a fixed value of k, to use non-fixed number of nearest neighbours i.e. k. Large value of parameter k would also increase the computational cost and time in case of large data sets. To solve this problem it has applied three heuristics so that early break of the algorithm can be possible. These heuristics on fulfilment of a fixed condition would break out from the algorithm. This would save computational time of the algorithm.

E. KNN with Shared Nearest Neighbours
[21] proposed an another variant of KNN algorithm which uses shared nearest neighbours to classify documents. To find neighbours of a novel tuple, it uses BM25 similarity measure. A threshold is set, only that much number of nearest neighbours can vote for classification of an unknown tuple

F. KNN with K-Means
One of the shortcomings of KNN algorithm is its high computation complexity. [22] tried to alleviate this drawback by combining KNN algorithm with the clustering algorithm K-Means. In the proposed algorithm first the clusters of the different categories in training data set are formed. The centres of these newly formed clusters will now act as new training samples. To classify an unknown tuple the distance of it is computed with these new training tuples and with which the tuple has least distance it will be assigned to that class. The benefit of this variant of KNN is that there is no need of passing the input parameter k as we have to do in standard KNN.

G. KNN with Mahalanobis Metric
The performance of KNN algorithm largely depends on the distance metric which is used to find the distance between any two tuples. [23] introduces a new distance metric called Mahalanobis distance metric. It transforms the whole input space using linear transformation. In this transformed input space the Euclidean distance is same as Mahalanobis distance between any two data points. Euclidean distance is the distance between any two points whereas Mahalanobis distance is a distance between a point and a distribution. If the point represents the mean of the distribution then the Mahalanobis distance would be zero. The main benefit of taking Mahalanobis distance metric instead of Euclidean distance metric is that it also reckoned the correlation between data tuples [24].

H. Generalized KNN
KNN algorithm is not only used for prediction of categorical class attributes but continuous-valued class attributes also. In the later case, the average of the values of the class attribute is assigned to the class attribute of the unknown tuple. [25] implemented KNN algorithm for prediction of a continuous-valued class attribute.

I. Informative KNN
[26] introduces a new measure called informativeness. This measure takes into account the fact that all the k nearest neighbours are not equally important. The algorithm would take two input parameters instead of one i.e. k and I. the value of I will decide that how many informative data tuples are to be considered for classification of an unknown tuple. The proposed algorithm first finds the k nearest data tuples to the test tuple and after that it computes informativeness of these k data tuples. The majority class of the I most informative data tuples would be the class of unknown data tuple.

J. Bayesian KNN
Bayesian classifier is one of the classification algorithms which give quite good accuracy. Rather than giving the class of test tuple as output, the Bayesian classifier give the membership of the tuple in the classes in form of probabilities. To increase the performance of KNN algorithm in ranking, a combination of Bayesian classifier and KNN is proposed in [27]. In the proposed algorithm, initially the k nearest neighbours to the test tuples is determined. After that these k data tuples are used to train the classifier. The classifier would give membership probabilities of the test tuple in the classes as result. These probabilities are used for ranking instances.

K. SVM KNN
Support Vector Machine (SVM) is a classification technique which can be employed on both linear and non-linear data. [28] proposed a hybrid version of KNN with SVM for visual category recognition. In SVM-KNN algorithm, the k nearest neighbours to the unknown tuples is used to train SVM. For implementation of this hybrid algorithm, first the k nearest data tuples is determined. After that pair-wise distance among these k data tuples is

computed. Kernel matrix is computed from this obtained distance matrix. This computed kernel matrix is given as input to SVM classifier. The output would be the class of unknown tuple.

L. Some other Extensions for KNN

[29] proposed extensions for KNN algorithm which are density-based KNN classifier, variable k KNN classifier, weighted KNN classifier, class-based KNN classifier, discernibility KNN classifier. Instead of just counting the number of neighbours the density-based KNN classifier takes into account one another factor i.e. density. The variable k KNN classifier selects different values of k for different training data sets. Weighted KNN classifier computes weights for all the features of data set. Class-based KNN classifier selects different k for distinct classes depending on the number of tuples it contains. Discernibility KNN classifier uses the concept of discernibility which measures how the distinct classes of a data set are easily distinguished.

IV. KNN WITH EVOLUTIONARY COMPUTATION

Soft computing in integration with data mining techniques can be used for knowledge discovery in efficacious manner. Soft computing contrary to hard computing gives approximate solution to the problems for which there exists no exact mathematical model [30]. The principal components of soft computing include Neural Network, Fuzzy Logic, Rough Sets and Evolutionary Computing. These soft computing techniques include system structure and computational method [31]. The soft computing introduces the factor of optimization in data mining techniques when combined together. The impact and role of soft computing in data mining can be seen in [32],[33],[34],[35]. Now the question arises that why we need evolutionary computation to inculcate with data mining when there exists already a number of data mining techniques for knowledge discovery. The traditional data mining techniques are designed to give accurate results which in turn increase the computational cost. Also more time gets elapsed to produce the results. Because the result produced by these traditional data mining techniques is very sensitive to noise, these tend to produce an over fit model. The another problem with the traditional data mining techniques is that a priori knowledge is assumed by them regarding the data set if it is not available. Opposite to traditional data mining techniques, the evolutionary computing produces approximate results and is applicable to any field. Additionally, it reduces the computation cost and the time elapsed for producing the results. Adding to the pros the evolutionary computing is noise insensitive. The attribute interaction in the evolutionary computing is better than that of conventional data mining techniques. Incorporating evolutionary computing with data mining techniques will dispose of the problems associated with data mining techniques. There are various data mining tasks which can be formulated as optimization problems such as classification, feature selection, clustering etc. There are two types of algorithms involved in evolutionary computing: evolutionary algorithms and metaheuristic algorithms. These algorithms are basically inspired by natural biological processes. Both of these algorithms can be desegregated with KNN algorithm to enhance its performance. Genetic algorithm and differential algorithm are two evolutionary algorithms. Ant colony optimization, particle swarm optimization, firefly algorithm, cuckoo search, tabu search are example of metaheuristic algorithms. In previously done studies many researchers have combined KNN with evolutionary and metaheuristic algorithms to overcome its limitations. Some of the previously done studies are mentioned below:

A. Fuzzy KNN

The membership of a novel tuple assigned by the KNN algorithm is the crisp one. [36] proposed fuzzy version of KNN. Instead of giving crisp membership of the tuples it gives fuzzy membership.

B. Fuzzy-Rough KNN

This version of KNN combines fuzzy set and rough set theory. [37] proposed this variant of KNN. It computes the lower and upper approximations for each distinct class. The classification of unknown tuples is based on their membership to these approximations.

C. KNN with Genetic Algorithm

The main limitations of KNN are: its classification procedure solely dependent on the training data set, all the tuples are treated equally and the last one is its high complexity due to computation. [38] proposed a hybrid version of KNN with genetic algorithm. Instead of using a distance metric to find the k nearest training tuples to the test tuple, genetic algorithm is used to find k-nearest neighbors in the proposed algorithm. It has considerably improved the performance of KNN algorithm and also has reduced the computational cost to a large extent.

Another variant of KNN indulging genetic algorithm in it is proposed in [39] for diagnosing heart disease. This combination was proposed to enhance the performance of KNN in diagnosis of heart disease. There may be irrelevant and redundant features present in the data set which may badly influence the results of KNN. In the proposed algorithm, the genetic algorithm is used to rank the attributes. The low ranked attributes are discarded from the process of classification before applying KNN. Then this pruned data set is used for classification of a novel tuple. Its classification is done as we normally do using standard KNN.

D. KNN with Ant Colony Optimization(ACO)

To get good results from KNN it is really important to select good features from the existing ones in the data set. The standard KNN algorithm takes all the features with same weight which can negatively affect the results. [40] combines multiple KNN classifiers for which features are selected using ACO algorithm. Each KNN classifier uses different combination of features. As the name indicates ACO algorithm is based on the behavior of ants.

 DOI 10.17148/IJARCCE.2016.55101

E. KNN with Particle Swarm Optimization(PSO)

PSO algorithm is based on how the swarms behave. There are two variants of KNN with PSO available in previous studies. The first one is given by [41] in which this combination is used for diagnosis of coronary artery disease. In this, the representatives of all the distinct classes in the data set are determined using PSO algorithm and after that the classification of novel tuples are done using KNN algorithm. This reduces the cost of computation considerably. The second one is given in [42]. Every feature in the data set is not equally important as mentioned before. So, this study proposed to use PSO algorithm to compute weights for all the features in the data set and then accordingly use those features for classification using KNN.

F. KNN with Artificial Bee Colony Algorithm (ABC)

Similar to PSO-KNN algorithm variants described just above ABC-KNN also has been implemented in two different manners. The first one is for assigning weights to the features of data set using ABC algorithm which is based on foraging behavior of honey bees. This one is proposed in [43]. And the second one is proposed in [44] for diagnosis of coronary artery disease using the same concept as done in case of PSO-KNN. It first computes the representatives of the distinct classes in the data set using ABC algorithm and after that the classification of unknown tuples is accomplished using standard KNN algorithm. Now, for classification the distance of the unknown tuple is needed to be computed with those representatives only. In the later one there is no need of passing input parameter k as we do in case of standard KNN algorithm.

V. CONCLUSION

After studying KNN algorithm thoroughly we witnessed that there are many limitations associated with it despite of a number of pros. Many researchers have employed their own perception to unriddle these limitations. On analyzing the studies cited above we came to know that each of the modified KNN algorithm give satisfactorily good results when compared to the results given by standard KNN. The proposed modified KNN algorithms can be broadly classified into two categories. The first category encompasses those variants of KNN that tried to improve the performance of KNN by making alterations to the result influencing factors for KNN i.e. deciding appropriate value of KNN, distance metric to be used to measure similarity between data tuples, weights assigned to the attributes of data set. The second category embraces all those hybrid versions of KNN that has infused evolutionary computing. None of these modified KNN algorithms succeeded in eradicating all the limitations of standard KNN. But the result of these modified KNN algorithms is considerably good. Locally adaptive KNN, Adaptive KNN, KNN with shared nearest neighbors ameliorate the results of KNN by using different methodologies to determine the value of k. The performance of KNN also relies on the type of distance metric used. Generally Euclidean distance metric is used in KNN but KNN with the two new proposed distance metrics i.e. Mahalanobis and informativeness have also performed substantially well. The weight adjusted KNN also give better accuracy to than that of standard KNN by assigning different weights to the attributes involved. KNN performs equally well in case of predicting continuous valued class attribute too.

KNN when combined with evolutionary computing algorithms i.e. genetic algorithm, ant colony optimization (ACO), particle swarm optimization (PSO), artificial bee colony algorithm (ABC), give considerably good results. In studies cited above, genetic algorithm is used for feature selection and to determine k nearest data points to the unseen tuple. ACO and PSO metaheuristic algorithms have been used for extracting good features of data set for KNN whereas ABC algorithm is used for assigning weights to the attributes of data set according to their importance. So in the end we can conclude that the performance of standard KNN can be improved to a great extent by making modifications to it.

REFERENCES

[1] J. Han, M. Kamber, and J. Pei, Data mining: concepts and techniques. Elsevier, 2011.
[2] “Data mining,” Wikipedia, the free encyclopedia. 03-May-2016.
[3] “Data Mining Applications,” ZenTut. .
[4] S. Bagga and G. N. Singh, “Applications of Data Mining,” Int. J. Sci. Emerg. Technol. Latest Trends, vol. 1, no. 1, pp. 19–23, 2012.
[5] N. Padhy, D. P. Mishra, and R. Panigrahi, “The Survey of Data Mining Applications And Feature Scope,” Int. J. Comput. Sci. Technol., vol. 2, no. 3, pp. 43–58, Jun. 2012.
[6] B. M.Ramageri, “DATA MINING TECHNIQUES AND APPLICATIONS,” Int. J. Comput. Sci. Eng., vol. 1, no. 4, pp. 301–305.
[7] R. Kumar and R. Verma, “Classification algorithms for data mining: A survey,” Int. J. Innov. Eng. Technol. IJIET, vol. 1, no. 2, pp. 7–14, 2012.
[8] S. S. Nikam, “A Comparative Study of Classification Techniques in Data Mining Algorithms,” Orient. J. Comput. Sci. Technol., vol. 8, no. 1, pp. 13–19, Apr. 2015.
[9] “A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm,” God, Your Book Is Great !!, 18-May-2010. .
[10] S. Bafandeh Imandoust and M. Bolandraftar, “Application of K-Nearest Neighbor (KNN) Approach for Predicting Economic Events: Theoretical Background,” Int. J. Eng. Res. Appl., vol. 3, no. 5, pp. 605–610, Oct. 2013.
[11] M. Shouman, T. Turner, and Stocker, “Applying k-nearest neighbour in diagnosing heart disease patients,” Int. J. Inf. Educ. Technol., vol. 2, no. 3, p. 220, Jun. 2012.
[12] Y. Liao and V. R. Vemuri, “Using K-Nearest Neighbor Classifier for Intrusion Detection.”
[13] K. Alkhatib, H. Najadat, I. Hmeidi, and M. K. A. Shatnawi, “Stock Price Prediction Using K-Nearest Neighbor (kNN) Algorithm,” Int. J. Bus. Humanit. Technol., vol. 3, no. 3, pp. 32–44, Mar. 2013.
[14] G. Toker and O. Kirmemis, “Text Categorization using k Nearest Neighbor Classification,” Surv. Pap. Middle East Tech. Univ.
[15] X. Geng, T.-Y. Lui, T. Qin, A. Arnold, H. Li, and H.-Y. Shum, “Query dependent ranking using k-nearest neighbor,” in Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, 2008, pp. 115–122.
[16] D. T. Larose and C. D. Larose, Data Mining and Predictive Analytics, 2nd ed. Wiley, 2015.
[17] D. Wettschereck and D. Thomas G., “Locally adaptive nearest neighbor algorithms,” Adv. Neural Inf. Process. Syst., pp. 184–184, 1994.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

[18] H. E.H.S, E.-H. Sam, G. Karypis, and V. Kumar, "Text categorization using weight adjusted k-nearest neighbor classification," in Text categorization using weight adjusted k-nearest neighbor classification, Springer Berlin Heidelberg, 2001, pp. 53–65.
[19] B. Li, S. Yu, and Q. Lu, "An Improved k-Nearest Neighbor Algorithm for Text Categorization," in Proceedings of the 20th International Conference on Computer Processing of Oriental Languages, Shenyang, China, 2003.
[20] S. Ougiaroglou, A. Nanopoulos, A. N. Papadopoulos, Y. Manolopoulos, and T. Welzer-Druzovec, "Adaptive k-Nearest-Neighbor Classification Using a Dynamic Number of Nearest Neighbors," in Advances in Databases and Information Systems, Y. Ioannidis, B. Novikov, and B. Rachev, Eds. Springer Berlin Heidelberg, 2007, pp. 66–82.
[21] Y. Cai, D. Ji, and D. Cai, "A KNN Research Paper Classification Method Based on Shared Nearest Neighbor," in Proceedings of the 8th NTCIR Workshop Meeting, 2008.
[22] P. WiraBuana, S. Jannet D.R.M., and I. Ketut Gede Darma Putra, "Combination of K-Nearest Neighbor and K-Means based on Term Re-weighting for Classify Indonesian News," Int. J. Comput. Appl., vol. 50, no. 11, pp. 37–42, Jul. 2012.
[23] K. Q. Weinberger and L. K. Saul, "Distance Metric Learning for Large Margin Nearest Neighbor Classification," J. Mach. Learn. Res., vol. 10, pp. 207–244, Dec. 2009.
[24] "Mahalanobis distance," Wikipedia, the free encyclopedia. 04-Mar-2016.
[25] M. Sharma and S. Sharma, "Generalized K-Nearest Neighbour Algorithm- A Predicting Tool," Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 3, no. 11, Nov. 2013.
[26] Y. Song, J. Huang, D. Zhou, H. Zha, and C. L. Giles, "Iknn: Informative k-nearest neighbor pattern classification," in Knowledge Discovery in Databases: PKDD 2007, 2007, pp. 248–264.
[27] L. Jiang, H. Zhang, and J. Su, "Learning k-nearest neighbor naive bayes for ranking," in Advanced Data Mining and Applications, 2005, pp. 175–185.
[28] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition," in Computer Vision and Pattern Recognition,2006 IEEE Computer Society Conference on, 2006, pp. 2126–2136.
[29] Z. Voulgaris and G. D. Magoulas, "Extensions of the K Nearest Neighbour Methods for Classification Problems," in Proceedings of the 26th IASTED International Conference on Artificial Intelligence and Applications, Anaheim, CA, USA, 2008, pp. 23–28.
[30] "Soft computing," Wikipedia, the free encyclopedia. 11-Mar-2016.
[31] Y. Kumar and D. Kumar, "Parametric Analysis of Nature Inspired Optimization Techniques," Int. J. Comput. Appl. Found. Comput. Sci. N. Y. USA, vol. 32, no. 3, pp. 42–49, Oct. 2011.
[32] S. Mitra, S. K. Pal, and P. Mitra, "Data mining in soft computing framework:a survey," IEEE Trans. Neural Netw., vol. 13, no. 1, pp. 3–14, 2002.
[33] K. Lal, N. . Mahanti, and P. Bihar, "Role of soft computing as a tool in data mining," pp. 526–537, 2011.
[34] Y. K. Mathur and A. Nand, "Soft Computing Techniques and its impact in Data Mining," Int. J. Emerg. Technol. Adv. Eng., vol. 4, no. 8, Aug. 2014.
[35] R. Kruse, C. Borgelt, D. D. Nauck, N. J. Van Eck, and M. Steinbrecher, "The Role of Soft Computing in Intelligent Data Analysis," in Proceedings of the 16th IEEE International Conference on Fuzzy Systems, 2007, pp. 9–17.
[36] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy K-nearest neighbor algorithm," IEEE Trans. Syst. Man Cybern., vol. SMC-15, no. 4, pp. 580–585, Jul. 1985.
[37] R. Jensen and Cornelis, Chris, "Fuzzy-rough nearsest neighbor classification," in Transactions on rough sets XIII, Springer Berlin Heidelberg, 2011, pp. 56–72.
[38] N. Suguna and K. Thanushkodi, "An Improved k-Nearest Neighbor Classification Using Genetic Algorithm," Int. J. Comput. Sci. Issues, vol. 7, no. 4, pp. 18–21, Jul. 2010.
[39] M. A. Jabbar, B. Deekshatulu, and P. Chandra, "Classification of heart disease using K-nearest neighbor and genetic algorithm," Procedia Technol., vol. 10, pp. 85–94, Dec. 2013.
[40] Shrivastava, Shailendra Kumar and P. Mewada, "ACO Based Feature Subset Selection for Multiple K-Nearest Neighbor Classifiers," Int. J. Comput. Sci. Eng., vol. 3, no. 5, pp. 1831–1838, May 2011.
[41] I. Babaoğlu, O. Findik, E. Ulker, and N. Aygul, "A novel hybrid classification method with particle swarm optimization and k-nearest neighbor algorithm for diagnosis of coronary artery disease using exercise stress test data," Int. J. Innov. Comput. Inf. Control, vol. 8, no. 5, 2012.
[42] M. CHEN, J. GUO, C. WANG, and Fenlin WU, "PSO-based Adaptively Normalized Weighted KNN Classifier," J. Comput. Inf. Syst., vol. 11, pp. 1407–1415, Apr. 2015.
[43] H. Yigit, "ABC-based distance-weighted kNN algorithm," J. Exp. Theor. Artif. Intell., vol. 27, no. 2, pp. 189–198, Mar. 2015.
[44] İ. Babaoğlu, "Diagnosis of Coronary Artery Disease Using Artificial Bee Colony and K-Nearest Neighbor Algorithms," Int. J. Comput. Commun. Eng., pp. 56–59, 2013.

.