Big Data Classification using Fuzzy K-Nearest Neighbor

Malak El Bakry
Assistant Lecturer
Faculty of Computer Science
October University for Modern Sciences and Arts
Giza, Egypt

Soha Safwat
Lecturer
Faculty of Computer Science
October University for Modern Sciences and Arts
Giza, Egypt

Osman Hegazy
Professor
Faculty of Computers and Information,
Cairo University
Giza, Egypt

ABSTRACT

Because of the massive increase in the size of the data it becomes troublesome to perform effective analysis using the current traditional techniques. Big data put forward a lot of challenges due to its several characteristics like volume, velocity, variety, variability, value and complexity. Today there is not only a necessity for efficient data mining techniques to process large volume of data but in addition a need for a means to meet the computational requirements to process such huge volume of data. The objective of this paper is to classify big data using Fuzzy K-Nearest Neighbor classifier, and to provide a comparative study between the results of the proposed systems and the method reviewed in the literature. In this paper we implemented the Fuzzy K-Nearest Neighbor method using the MapReduce paradigm to process on big data. Results on different data sets show that the proposed Fuzzy K-Nearest Neighbor method outperforms a better performance than the method reviewed in the literature.

Keywords

Big data; Classification; Fuzzy k-nearest neighbor; Fuzzy logic; Hadoop; MapReduce

1. INTRODUCTION

Various innovations of technology are driving the spectacular growth in data and data gathering, this is the reason behind the question of “why big data has become a recent area of strategic investment for its organizations?” [1].

Big data is a group of enormous volume of structured and unstructured data from different sources. The different sources of big data are such as data coming from social network, data generated by machine, and traditional enterprise. Big data is huge and difficult to develop using ordinary databases and software techniques. Due to its complexity it require a new architecture, techniques, algorithms, and analytics to manage it, read out the values and extract the hidden knowledge from it [1].

Now it is impossible for analysts to extract a meaningful useful conclusion from data in a short time frame due to the huge volume of the data. So techniques of data mining are looked upon as tools that can be used to automate the process of knowledge discovery and define relationships and patterns of likeness given a completely random and raw data set. Unsupervised data are majority of data collected for analysis. This show that there is a need to an effective technique that can process on such unsupervised data sets and convert what might seem to be totally random and meaningless into something more meaningful and valuable [2].

Accessing data, computing data, domain knowledge, privacy of data, and data mining are main problems of the big data [3]. Due to these challenges processing data and data mining techniques becomes a critically important role in development technology [2].

Classification is one of the most useful techniques in data mining that classify data into structured class or groups. It helps the user in discovering the knowledge and future plan. Classification supplies the user with an intelligent decision making. Classification consists of two phases; the first phase is learning process phase in which a huge training data sets are analyzed and then it create the patterns and the rules. The second phase is evaluation or testing and recording the accuracy of the performance of classification patterns. The purpose of classification is to be able to use its model to predict the class label of objects whose class label is unknown. Various forms can be represented for instance Neural Networks, classification rules, mathematical formulas or decision tree [1].

K Nearest Neighbor (KNN) is one of the most standout classification algorithms in data mining. It is based on homogeneity, which is drawing a comparison between the given test record with training records which are similar to it. K Nearest Neighbor classification provide us with the decision outline locally which was developed due to the need to carry out discriminate analysis when reliable parametric estimates of probability densities are unknown or hard to define. K is a constant pre-defined by the user. The testing data are classified by giving the label which is most frequent repeated among the k training samples nearest to that query point [4].

The Fuzzy KNN classifier works by assigning a membership value to the unlabeled signature that supply the framework with proper data for estimating the certainty of the decision. Each of the defined classes has a fraction of unlabeled signature defined by Fuzzy membership coefficient. Delineation means that the membership of two classes is relatively high and confusion means that the membership between two classes in very low. When assigning a non-fuzzy label to the signature the above data becomes very important [5].

Assigning an object to the unknown class is a greater advantage of using a Fuzzy system over crisp system. In addition, crisp system can assign the object to wrong class. The fuzzy k-NN classifier classifies the data according to the training data and the following fuzzy information taken from it. The user can indirectly control the defuzzification level to specify the percentage of wrong decision is “worth” to the process. In a lot of cases, setting the defuzzification level has more advantage than not setting it because when more defects are categorized as unknown is much better than classifying it wrong. This is so accurate in many cases where classifying a defect wrong could result in a stronger economic effect or income loss [5].

Several classification techniques using MapReduce architecture are implemented as the use of Linguistic Fuzzy rule by Victoria Lopez, Sara Del Rio, Jose Manuel Benitez and Francisco Herrera [11], the K-Nearest Neighbor Algorithm by Prajesh P Anchalia and Kaushik Roy [2], the Support Vector Machine by Ke Xu, Cui Wen, Qiong Yuan, Xiangzhu He, and Jun Tie [12], Neural Networks by Chetan Sharma [6].

In this paper we implemented the Fuzzy K-Nearest Neighbor method using the MapReduce paradigm to process on big data to enhance the performance of several methods that have been reviewed in the literature [7]. To do so, we rely on the success of the MapReduce framework. The map phase splits the data and the reduce phase perform join to all outcome from the mappers and gives us the final output. To test the performance we conducted experiment on different data sets. The experimental study indicates an analysis of the accuracy of the testing.

The rest of this paper is organized as following section II Related work. Next section III contains big data classification. The proposed system is shown in section IV. The experimental results are then discussed in section V. Finally section VI concluded the work and showed the future work.

2. RELATED WORK

Due to the huge increase in the size of the data and the great amount of information that are added every second the word big data became one of the most important words nowadays. It becomes troublesome to perform efficient analysis using the current traditional techniques on the big data. So, big data put forward a lot of challenges. The MapReduce paradigm has become one of the most challengeable areas in this era due to its infrastructure and importance in dealing with big data. Also the Hadoop have gained a lot of attention because it’s an open source and can deal with big data. According to the importance of the big data classification there are a lot of related works and I will list some of them below.

The k-Nearest Neighbor method is placed in the top ten data mining techniques. Prajesh P, Anchalia, and Kaushik Roy used this well-known classification technique to classify big data. They run this method on an apache Hadoop environment that of course uses the MapReduce paradigm to process on big data. They faced a lot of problems and the most important one is balancing between the friendly user interface and performance. They implemented the k nearest neighbor on the Hadoop using multiple computers to delete the limitations of computational capability and speeding up the processing time. This is done by having groups of systems working together and connected over a network. They also compared their results using a MapReduce K Nearest Neighbor with sequential K Nearest Neighbor and concluded that the MapReduce k nearest neighbor gives better performance than the sequential K Nearest Neighbor with big data [2].

Nasullah Khalid Alham, Maozhen Li, Yang Liu, and Suhel Hammoud used a MapReduce support vector machine technique. They named this technique MRSMO technique (MapReduce based distributed SVM algorithm for automatic image annotation). Their technique depends on partitioning the training data into subsets and sent these subsets across groups of computers. They evaluated their technique in an experimental environment and it result in a significant reduction in the training time and a high accuracy in both binary and multiclass classification [8].

Ke Xu , Cui Wen , Qiong Yuan, Xiangzhu He , and Jun Tie used the parallel Support Vector Machine based on MapReduce method for classification of emails which is a big data set. They implement an experiment on this data set and used many techniques in evaluation but the support vector machine based on MapReduces show a significant reduction in the training time. Big data sets are very complex to be analyzed using classical Support Vector Machine but the parallel Support Vector Machine depending on MapReduce and can deal easily with big data. The MapReduce distribute the subsets of the training data among many nodes to improve the computation speed and improve the accuracy [9].

Now days, the mobile data set became one of the most challenging big data set due to its continuous production. Classification on this data requires high specifications because of its nature. This data have three main challenges; which are the difficulty in keeping both the accuracy and the efficiency, the huge increase in the load of the system, and the data noise. Zhiqiang Liu, Hongyan Li, and Gaoshan Miao used the Back Propagation Neural Network MapReduce technique to classify big data of mobiles. They implemented a technique called MBNN (MapReduce-based Back propagation Neural Network) to be used in classification of data. A lot of experiments are performed using the cloud computing platform and concluded that the MBNN have the characteristics of superior efficiency, good scalability and anti-noise [10].

Changlong Li, Xuehai Zhou, and Kun Lu Implemented an Artificial Neural Networks in MapReduce paradigm. They represented it to accomplish the parameter configuration automatically for MapReduce. Their technique can adjust its software configuration and hardware configurations to the system automatically giving the cluster, MapReduce job, and frameworks. Their technique also can determine the ideal configuration of the system in suitable time with the help of ANN. They experiment their technique and show that it result in a great influence in optimizing the system performance and speeding the system up [11].

3. BIG DATA CLASSIFICATION

3.1 Big Data

Big Data definition depends on who is describing it and in which context. So there is no comprehensive definition for it. However at common level almost every definition of big data summarizes the concept to huge and increasing data masses and the process of analyzing that data. The basis of competition, productivity enhancement and creating important value for the world economy is one of the reasons behind the use of big data. It also decreased waste and expanded the quality of items and services which leads big data to become the main reason behind making better decisions. The big data have 3 characteristics which are the three Vs. The three Vs are Velocity, Variety, and Volume. Volume defines the large quantity of data produced periodically. Velocity is the speed at which data is generated. It should have high rapidity data. Variety is the various data formats, types, and structures. The type of data may cover various varieties such as Text, numbers, photos, video, audio, time series, and social media data. It may also include static data and dynamic data. Collecting many types of data can be generated by a single application. All these types of data need to be attached together to extract the knowledge. Extracting information will need advanced method to process due to its high volume, velocity, and variety [1].

3.2 The MapReduce paradigm

MapReduce is published by Google as a programming model for conducting difficult combination of huge amount of data [1]. The aim of the MapReduce program is to process on massive volume of data on multiple machines. MapReduce divides data into independent blocks. Due to this division a single task is divided into multiple subparts each one is handled by an independent node [12]. MapReduce are divided into two different steps; Map and Reduce. Each step is done parallel on sets of <key,value> pairs [2].

The Map function takes the divided data values as input, then it perform any function to every value in the input set and produces an output set. The output of the map is in the form of <key,value> pairs stored on each node. The output of the Map function is stored and sent to the Reducer [2].

The Reduce function takes the output from the Map function as an input for it and then generates another set of <key,value> as final output. The Reducer can't start until all the Map stage is finished and the results are sent to the appropriate machine.

The MapReduce framework consists of a single Master Job Tracker and multiple Task Trackers. The task tracker can be any node in the cluster. The Master Job Tracker is responsible for division of the input data, task scheduling, failure of the machine, re-execution of un-succeeded tasks, inter-machine communications and task status monitoring. The task tracker is responsible for executing the tasks assigned by the master. There is a file system to store both input and output files. The single Job Tracker can be a single point failure in this framework. MapReduce is an appropriate technique to deal with huge datasets and therefore ideal for mining Big Data of petabytes size that do not fit into a physical memory [2].

The MapReduce algorithm is implemented using many commercial and open-source technologies as a part of their internal architecture. One of the most efficient and popular techniques of implementing the MapReduce is the Apache Hadoop. Hadoop aimed to be used for data executing in a distributed computing environment. Any programming language can be used in implementing the MapReduce algorithm [12].

3.3 The Hadoop

ApacheTM Hadoop is an open source application that runs on a distributed computing environment and supports processing of Big Data with huge volume. It has been evolved from The Google File System[2].The architecture of the Hadoop contains many components such as the master server which directs jobs to the machines of the underlying worker, It also contains a package of components usually named the "Hadoop Common Package." This package consists of components for example the "Hadoop Distributed File System" (HDFS), MapReduce engine, and scripts to run the Hadoop installation [12].

HDFS contains name nodes and data nodes; they are downloaded on the server of the master and workers separately. The name node is responsible for managing data using data nodes located in the machines of the workers and mapping those files to data nodes. The data nodes on the worker computers implement read and write requests as required [2].

Data in the Hadoop framework is saved across multiple nodes in the HDFS. Replication of data in HDFS is done three times on separate nodes to provide protection from failure. To insure data integrity a checksum for the data blocks is continuously calculated. The programming libraries are used to perform the distributed computing tasks, which implement the MapReduce algorithm. All of these components work with each other to process on Big Data in a batch processing mechanism [12]. Due to the characteristics of Hadoop it achieves its goal of processing by breaking down the given data set and process on it individually at separate nodes that are connected together. Data will be subject to failure because of the distribution of it on multiple nodes. So, when a failure is detected the process can be restarted automatically again on another node. It also creates a copy of missing data from the available replicas of data. There is a failure point when only one single Namenode is available [2].

There are numerous open source projects based on the top of Hadoop such as Hive, Pig, Mahout, and Pydoop. Hive is a data warehouse framework that analyzes complex data. Pig is a dataflow framework that produces a series of MapReduce programs. Mahout is responsible for the machine learning libraries that focus on clustering, classification, frequent item-sets mining and evolutionary programming. Pydoop is a python package that support API for Hadoop MapReduce and HDFS [2].

3.4 Fuzzy Classification

Fuzzy logic is a technique of computing which is based on the "degree of truth" not like the traditional "true or false" (1 or 0) techniques. Fuzzy logics are the basics of the modern computer. The first one who invented the idea of Fuzzy logic was Dr. Lotfi Zadeh from the University of California at Berkeley in the 1960s. Solving the problem of the computer understanding of natural language was the problem that leads Dr. Zadeh to think in the Fuzzy logic. Absolute terms (0 and 1) are very difficult to be used in describing the natural language. It's very difficult to translate the natural language to 0 and 1. Fuzzy logic includes two extreme cases of the truth which are zero and one but also includes the different cases of truth in between, for example, the output of a comparison between two things could be not "tall" or "short" but ".38 of tallness." [13].

The Fuzzy KNN classifier works by assigning a membership value to the unlabeled signature that supply the framework with proper data for estimating the certainty of the decision. Each of the defined classes has a fraction of unlabeled signature defined by Fuzzy membership coefficient. When we assign the record to unknown class this is one of the main advantages of the Fuzzy logic over the crisp logic system. The Fuzzy KNN classifier classifies the testing data according to two things; the Fuzzy information taken from the training data and the training data itself [5].

Fuzzy KNN classifier was designed by Keller et al. [16], where class memberships are given to the sample, as a function of the sample's distance from its K Nearest Neighboring training samples. A Fuzzy K-NN Classifier is one of the most successful techniques for applications due to its simplicity and also because of giving some information about the certainty of the classification decision. Keller et al assume that the improvement of the error rate might not be the major advantage from using the FKNN model. More importantly, the model offers a percentage of certainty which can be used with a "refuse-to-decide" option. Thus objects with overlapping classes can be detected and processed individually. The Fuzzy KNN algorithm is shown below [14].

Algorithm 1 " Fuzzy K-Nearest Neighbor Classifier

```
1: Find the k- nearest neighbors of a sample $x$ .
2: Assign an input $x$ a membership vector "Soft labels".
        $(\mu(x) = [\mu_{c1}(x), \dots, \mu_{ci}(x), \dots \dots \mu_{cl}(x)])$
    $\mu_{ih}(x) = \mu_i(x_i) = \left\{ \begin{array}{l} 0.51 + \left(\frac{n_i}{k}\right) * 0.49, if\ c(x_j) = i \\ \left(\frac{n_i}{k}\right) * 0.49, if \qquad c(x_j) \neq i \end{array} \right\}$
3: Calculate Membership function
        $\mu_i(x) = \frac{\sum_{j=1}^{k} \mu_{ij}(1/(\| x - x_j \|^{2/(m-1)})}{\sum_{j=1}^{k}(1/(\| x - x_j \|^{2/(m-1)})}$
4: the class label is max $u_i(x)$
```

4 PROPOSED ALGORITHM

In this section we show the details for the proposed method to classify big data using Fuzzy K-Nearest Neighbor algorithms.

We divide the data sets to training data sets and testing data sets. The training data sets are 75% of the whole data, and the rest 25% of the whole data are the testing data sets. MapReduce divides data into independent chunks and the size of division is a function of the size of data and number of nodes available. As shown in Figure 1 the data sets are divided on several mappers using the map function. Each mapper contains the same number of samples. The reduce part takes the results of individual mappers and combines them to get the final result. The idea of the model is to build individual classifier on each group. Each classifier is used to classify the testing data and send the class label to the reducer function, then the reducer take the majority vote to decide the final class label for the testing data.

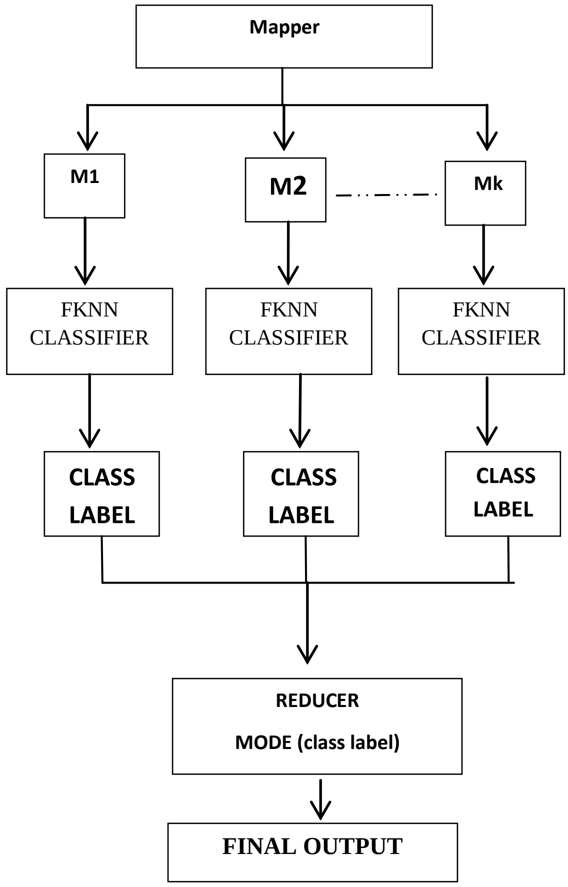

Fig.1 Proposed System

4.1 Dataset

In this experimental study we will use big classification data sets taken from the UCI repository. Table1 summarizes the main characteristics of these datasets. For each dataset, we show the number of records, number of attributes and number of classes. [15]

Table1. Data set description

Data	No of records	No of attributes	No of classes
Covtype-2	581012	54	2
Poker-2	1025009	10	2

4.2 MapReduce using Fuzzy KNN

After finishing organizing our training and testing data, we can apply the MapReduce Fuzzy K-Nearest Neighbor technique in a distributed environment by the algorithm discussed below.

Algorithm 2 :MapReduce design for FKNN

```
Read the Training data
ds=datastore('TrainData.csv','DatastoreType', 'tabulartext');
Procedure FKNN MapDesign
1:Load testing data file
Load TestFile
2: Create a matrix contains testing data
TestData = csvread(TestFile')
3:Read Samples from  TestData one at a time
While Not End Of File
4: Read Mappers from TrainData one at a time
While Not End Of Mappers
```

```
5: Call FKNN algorithm
Result = fknn('TrainData,Desird,train, TestData,K,M);
6: Send Result to the Reducer Design
add(intermediateValuesOut,Result);
End While
End While
call reducer
end procedure
Procedure FKNN ReducerDesign
Load Value of K
Load value of M
1:Load testing data file
TestData = csvread(TestFile')
2: Create a Vector contains testing data class label
3: Create a Matrix Outputs  M × D, M is number of mappers
and D is number of samples in TestFile
4: Initialize Outputs Matrix for all class labels
SET counters to ZERO
Outputs =zeros(D,M)
5: Read Result from MAP functions one at a time
while hasnext(intermediateValuesIn)
6: Write The output from MAP function into Matrix
output(i)=  Result
Assign the class label with the highest count for the testData
sample
SampleOutput= mode(output)
Mode: is the majority vote
End While
end procedure
```

As shown in the MapReduce design for FKNN, Map and Reduce are two different steps. Each step is done parallel on sets of <key,value> pairs. So the programs are bifurcated into Map and Reduce stage. Each Mapper function takes its training data after dividing it on the set of mappers, and also takes the testing data. The Map routine performs the FKNN function of training which is calculating the distance of each data point with the classes, then list out the class of the unknown data. After that the class label is sent to the reducer function by all mapper, and then the reducer function uses the majority vote function to classify the testing samples.

5 EXPERIMENTAL RESULTS

We compare the performance of the proposed system with a paper reviewed in the literature review which used Linguistic Fuzzy Rule [16].

Table2. Results

Methods Data Sets	FKNN	FRBCS
Covtype-2	75	74.96
Poker-2	69.0631	60.35

As shown in table 2 and figure 2, we compare the results of the proposed model and a method mentioned in literature on the same data set. We observed that a FKNN classifier gave a high accuracy comparing to the results when we used the FRBCS reviewed in the literature review when using the Poker data set and gave a slightly higher accuracy than the FRBCS using the Covtype data set.

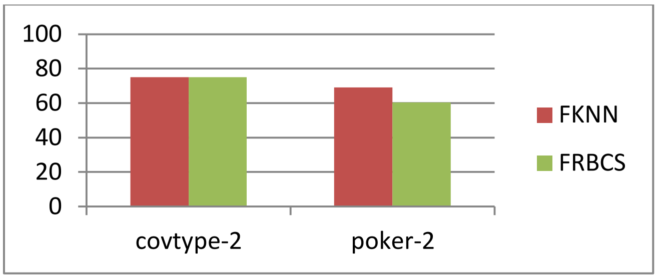

Fig.2 Comparison between the proposed system and literature reviewed FRBCS

6 CONCLUSION

In this study we introduced a comparative study of classification of big data. We used Fuzzy K-Nearest Neighbor classifier using the MapReduce paradigm. The proposed algorithm consists of two parts; the mapper and the reducer. The mapper algorithm is used to divide the data sets in to chunks over the computing nodes and produce a set of intermediate records. These records produced by the map function take the form of a “(key, data)” pair. Mapper in the individual nodes execute the computing process and send the results to the reduce function. The reducer algorithm receives the results of individual computations and put them together to obtain the final result. Good accuracy of the performance was obtained using the Fuzzy K-Nearest Neighbor method. Future work will concentrate on enhancing the results using Fuzzy techniques in the reducer rather than using the mode.

7 REFERENCES

[1]. S Mitha T, MCA, M.Phil, & M.Tech, V. (2013). Application of Big Data in Data Mining. International Journal of Emerging Technology and Advanced Engineering, 3(7), 390-393.

[2]. P Anchalia, Prajesh, and Kaushik Roy. The K-Nearest Neighbor Algorithm Using MapReduce Paradigm. Fifth International Conference on Intelligent Systems, Modelling And Simulation. 2014. Web. 15 Oct. 2015.

[3]. Koturwar, P., Girase, S., & Mukhopadhyay, D. (2015). A Survey of Classification Techniques in the Area of Big Data.

[4]. Pakize, S., & Gandomi, A. (2014). Comparative Study of Classification Algorithms Based On MapReduce Model. International Journal of Innovative Research in Advanced Engineering (IJIRAE), 1(7), 251-254.

[5]. Tobin, K., Gleason, S., & Karnowski, T. (n.d.). Adaptation Of The Fuzzy K-Nearest Neighbor Classifier For Manufacturing Automation.

[6]. Sharma, C. (2014). Big Data Analytics Using Neural networks.

[7]. Río, S., López, V., Benítez, J., & Herrera, F. (2015). A MapReduce Approach to Address Big Data Classification Problems Based on the Fusion of Linguistic Fuzzy Rules. International Journal of Computational Intelligence Systems, 422-437.

[8]. Nasullah Khalid Alham, Maozhen Li, Yang Liu, and Suhel Hammoud, (2011). a MapReduce-based distributed SVM algorithm for automatic image annotation

[9]. Xu, K., Wen, C., Yuan, Q., He, X., & Tie, J. (2014). A MapReduce based Parallel SVM for Email Classification. Journal of Networks JNW.

[10]. Zhiqiang Liu; Hongyan Li ; Gaoshan Miao.MapReduce-based Backpropagation Neural Network over large scale mobile data

[11]. Changlong Li1, Xuehai Zhou1, Kun Lu1. Implementation of Artificial Neural Networks in MapReduce Optimization.

[12]. Bhagattjee, B. (2014). Emergence and Taxonomy of Big Data as a Service.

[13]. Wu, X., Zhu, X., Wu, G., & Ding, W. (n.d.). Data mining with big data. IEEE Trans. Knowl. Data Eng. IEEE Transactions on Knowledge and Data Engineering, 97-107.

[14]. Keller, J.M., Gray, M.R., and Given, J.A., (1985). A Fuzzy K-Nearest Neighbor Algorithm. IEEE Trans. Syst., Man, Cybern., Syst., 15 (4), 580-585.

[15]. MRPR: A MapReduce solution for prototype reduction in big data classification Neurocomputing, Vol. 150 (February 2015), pp. 331-345, by Isaac Triguero, Daniel Peralta, Jaume Bacardit, Salvador García, Francisco Herrera

[16]. Río, S., López, V., Benítez, J., & Herrera, F. (2015). A MapReduce Approach to Address Big Data Classification Problems Based on the Fusion of Linguistic Fuzzy Rules. International Journal of Computational Intelligence Systems, 422-437.