

# Machine Learning in Apache Spark Environment for Diagnosis of Diabetes

Farshid B. Saravi<sup>b</sup>, Shadi Moghanian<sup>a</sup>, Giti Javidi<sup>c</sup>, Ehsan O. Sheybani<sup>c,\*</sup>

<sup>a</sup> Computer Science Department, Universidad Politécnica de Cataluña, Barcelona, Spain

<sup>b</sup> Research Consultant, CS-IT Hub, Florida, USA

<sup>c</sup> Muma College of Business, University of South Florida, Tampa, Florida, USA

Corresponding author: Ehsan O. Sheybani (e-mail: sheybani@usf.edu).

**Abstract**— Disease-related data and information collected by physicians, patients, and researchers seem insignificant at first glance. Still, the same unorganized data contain valuable information that is often hidden. The task of data mining techniques is to extract patterns to classify the data accurately. One of the various Data mining and its methods have been used often to diagnose various diseases. In this study, a machine learning (ML) technique based on distributed computing in the Apache Spark computing space is used to diagnose diabetes or hidden pattern of the illness to detect the disease using a large dataset in real-time. Implementation results of three ML techniques of Decision Tree (DT) technique or Random Forest (RF) or Support Vector Machine (SVM) in the Apache Spark computing environment using the Scala programming language and WEKA show that RF is more efficient and faster to diagnose diabetes in big data.

**Index Terms**— Diabetes, Diagnosis, Machine Learning, Wireless Body Area Networks, Apache Spark, Feature Selection.

## 1. INTRODUCTION

Nowadays, large volumes of data are generated in various applications that accumulate over time. To some extent, this makes extracting useful information from such large datasets more challenging. To provide a higher level of knowledge amidst massive amounts of data, data mining is considered an effective and practical method utilized by various techniques such as artificial neural network (ANN). Data mining can be a valuable technique in finding useful patterns and extracting knowledge from large datasets and databases (Das et al., 2018).

Data mining has great applications in healthcare using Machine Learning (ML) techniques. One of the essential IoT applications of machine learning (ML) in the medical field is to diagnose various diseases such as diabetes and heart disease. ML techniques can explore the pattern used in medical information and patient-related data in the diagnosis of diseases. Today, massive data (Big Data) is generated by Wireless Body Area Networks (WBANs) that can be used to diagnose all types of conditions. The data collected by WBANs are a valuable resource for pattern recognition in patients. However, these data need an appropriate platform for analysis to identify their hidden pattern. Detecting these hidden patterns are valuable to diagnosing illness. ML techniques for

processing large amounts of data related to patient information require an appropriate processing platform to detect disease patterns in real-time accurately. Diabetes is considered the disease of the present century due to lack of physical activity, poor diet, and genetic backgrounds. It is estimated that the number of people with diabetes worldwide has reached millions, and that number is increasing year by year (Wu et al., 2018).

Diabetes or hyperglycemia occurs when blood glucose levels rise, or the body cells are unable to produce the proper insulin to absorb the sugar in the cells. Increased blood sugar or glucose levels can cause many risks to the person. This can damage the arterial walls, make the patient prone to atherosclerosis, or negatively affect one's vision (Ruano et al., 2018).

The devastating effects of diabetes are not limited to the two cases mentioned above, it can also affect the kidneys or cause heart attack or stroke. In advanced stages, the organs are destroyed and cut off by the disease. The high cost of treatment for the disease has made it a costly illness for individuals. Early detection and reduction of its effects are of interest to researchers and medical professionals. Invasive criteria for diabetes diagnosis are of less interest to the ordinary public (Bequette et al., 2018).

One of the challenges with a diabetes diagnosis is the administration of clinical trials or blood tests (Bequette et al., 2018). Delay in the diagnosis of diabetes causes the disease to go through its latency without any specific symptoms. Its symptoms are known when it has caused various damages to the organs of the body. Diabetes has a set of early signs that can help researchers diagnose it. For example, people's weight and age are vital indicators in diagnosing this dangerous disease and various methods such as data mining and ML can help with the early diagnosis (Zarkogianni et al., 2018).

One way to gather information from diabetic patients is to use WBANs to collect individuals' biomarkers and send them to a base station using a set of body sensors (He et al., 2018). The volume of information received by these networks is the big data type. It needs to be processed in distributed systems to detect the disease pattern accurately in a short runtime (Tan et al., 2018).

One of the significant data processing architectures is the Apache Spark framework capabilities that perform bulk computing based on distributed systems. Spark can be

considered a distributed architecture in computer networks that uses computer systems to process a calculation. In this processing architecture, cloud space is applied to many computers interconnected to each other to perform high-speed heavy calculations (Ramirez-Gallego et al., 2018).

Apache Spark can be considered a cloud computing space that performs processing on several virtual machines or clusters. Apache Spark architecture distributes bulk processing onto network clusters. Each cluster or computer plays a part in the data processing. Apache Spark Processing Architecture is an ideal architecture for processing big data types where data can be treated, presenting a cloud computing space for users. Since the data collected by WBANs are of a big data type and have large scales, the diagnosis and runtime of the disease pattern are high. Therefore, it is necessary to provide suitable platforms for processing network information and analyzing such large disease-related datasets (Sarabia-Jacome et al., 2018).

The aim of this study is analyze large-scale diabetic patients' information using ML techniques such as DT, RF, and SVM distributed over a set of clusters so that each network cluster performs part of the ML process on input data. In order to do this, we use the Apache Spark framework.

This paper is organized as follows. Section II reviews the relevant literature on Machine learning techniques in big data processing, Apache Spark, WBANs, and the Internet of Things (IoT). Section III provides a learning framework for diabetes diagnosis in Spark computing space. Section IV discusses the experiment results of machine learning techniques. Finally, conclusions and future works are discussed in Section V.

## 2. RELATED WORK

### A. Learning methods

Learning methods are divided into three main categories of unsupervised learning, supervised learning, and semi-supervised learning. An indicator of a variety of learning methods is the presence or absence of a label on the data. Labeled data is said to have a set of output attributes or data class numbers and input and descriptive attributes and can be used for learning (Shickel et al., 2018).

The critical difference between supervised and unsupervised learning techniques is that in the unsupervised method, the data do not provide information about their class to the learning algorithm. Whereas, in the supervised technique, the data provide information about their class to the learning algorithm. Therefore, supervised methods have more learning power than unsupervised algorithms. Unlike supervised methods that require learning, unsupervised techniques try to identify patterns without any learning and training data. Unlike these learning methods, in semi-supervised learning, the data are labeled and unlabeled. A combination of these two types of data is used for this kind of education (Humayun et al., 2018).

Several learning methods are based on unsupervised and semi-supervised learning methods, but most learning methods use supervised algorithms. The clustering technique is an

example of the unsupervised methods. In contrast, a supervised technique can include artificial neural network (ANN), decision tree (DT), regression, support vector machine (SVM), support vector regression, and random forest (RF) (Dutt et al., 2017).

**Decision tree (DT).** The decision tree is one of the supervised learning methods that use a tree structure to classify information. In this classification structure, each tree node decides whether the sample is on its left or right. The decision tree has a set of leaves, and its values specify the class of the sample. Various algorithms, such as C4.5 and CRT (systems implemented in C for the UNIX environment for classification-based intelligent decision making), have been developed to build decision trees (Aich et al., 2018). An example of the decision tree application to diagnose diabetes with the help of the decision tree can be seen in Fig. 1:

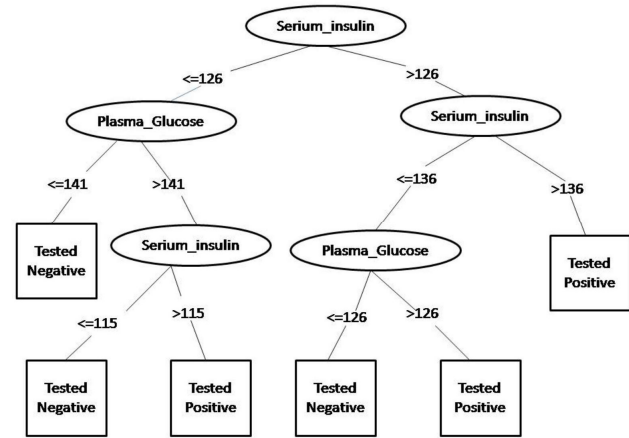


Fig. 1. Application of the decision tree in the diagnosis of diabetes

The diagram above Barale & Shirke, 2016) provides a tree structure based on the characteristics associated with diabetic patients, showing the disease status or health status of the individual in the leaf.

**Random Forest (RF).** Ensemble Method is an ML method that uses several learning and voting techniques between them for final classification. In this learning method, the training data are delivered to  $n$  classification techniques such as DT, RF, SVM, and so on. Each of them creates a classification model. The test data are given to each of these models, and voting specifies the final class as a sample. Voting mechanism means that all models focus on a consensus for a sample of class number. Finally, the majority of votes shows the selected final class number (Komal Kumar et al., 2019).

RF is one of the learning algorithms with a hybrid learning approach based on a hybrid learning technique. This learning method uses a set of decision trees to find the best classification. In this algorithm, many random decision trees are created. The best ones are used for classification at each stage. In this method, many random samples and attributes from the dataset are selected for each decision tree for categorization. The best decision tree with the least error in classification is considered as a model (Xu et al., 2020).

An example of the conceptual mechanism of a random forest learning technique that uses a set of random decision trees is shown in Fig. 2:

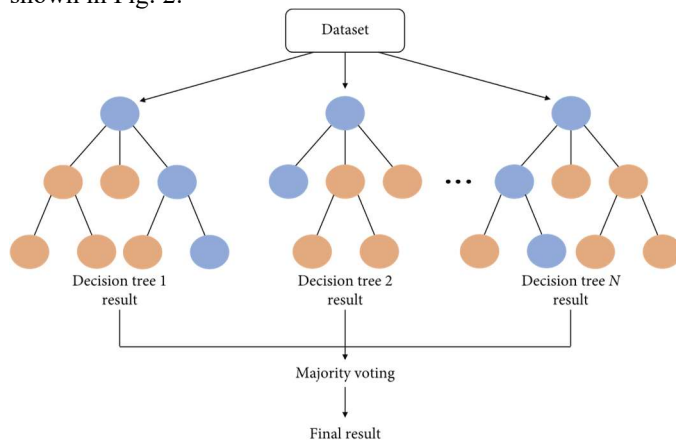


Fig. 2. Structure of a random forest sample (Jaehoon Kim et al., 2021)

**Support Vector Machine (SVM).** One of the supervised learning methods used for classification or regression is the SVM technique. The purpose of this technique is to reduce operational risk in classification or regression. In contrast, a method such as ANN seeks to reduce modeling error. The SVM technique and its classification application attempts to create a single line, plane, or super-plane in 2D, 3D, and over-three-dimensional space to separate the data. Then, two classes of data can be separated with the least possible interference. Data mining and ML of a line are placed with the margins between the two classes of data. It attempts to make the possible separation. In this learning method, the data, and examples from the two classes at the edge of the separator strip are called support vectors. Their role is to move the strip between the data to provide the most separation (Moreira et al., 2018).

An SVM problem-solving mechanism is shown in Fig. 3 to classify individuals into healthy and patient classes:

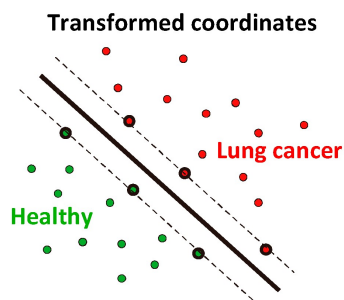


Fig. 3. Application of an SVM in Diagnosis (Sakumura et al., 2017)

#### B. Wireless Body Area Network (WBAN)

WBAN is a type of wireless sensor network in which each node has a tiny size installed on the patient body to record biomarkers of patients such as heart palpitations, blood pressure, blood sugar sends them to doctors. The sensors used in these types of networks have low radio beams and face many challenges in sending packets and routing them (Fernandes et al., 2018).

WBANs can also be considered a subset of IoT because sensors

can be defined as intelligent objects with the ability to connect to IoT. The wireless sensor network plays an essential role in the healthcare. For instance, it can be used to monitor patients' activities at home or at the hospital (Al-Janabi et al., 2017). Instead of keeping the patients at the hospital where they can face a high risk of nosocomial infections, patients can stay at home while receiving treatments. They can be connected to body sensors to send biological information to physicians. If a medical threat is detected, the physician can send required treatments or report emergencies. WBAN can also be used for monitoring and controlling the athlete's activities during exercise and analyzing the data sent from these sensors (Taralunga & Florea, 2021). Fig. 4 shows the structure of WBAN for diabetic patient data:

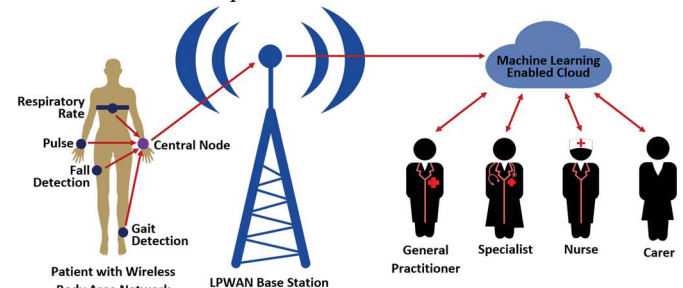


Fig. 4. An example of the use of a WBAN in the diagnosis of diabetes (Baker et al., 2017)

#### C. Internet of Things

Internet of Things (IoT) is a new technology that connects billions of smart devices and creates a huge network. IoT has been a popular and pioneering technology in the last decade. It has been able to link different objects together and increase their productivity. The concept of Things is a deep and broad term and can include a simple sensor or an industrial complex. Objects that can connect to IoT are devices such as watches, means of transportation, smart cars, industrial components (Al-Turjman & Deebak, 2021).

A variety of hardware has been introduced to implement the IoT model in the real world. For example, devices such as the Raspberry Pi, Arduino, Smartphone, Field Programmable Gate Array (FPGA), and Microcontroller have been introduced. Each of these hardware provides users with the opportunity to create their own IoT network or connect their intelligent objects to the network. Raspberry Pi is the best and most advanced of this hardware. The Raspberry Pi can be considered a device or a mini-computer for connecting sensors and objects to it. The hardware does not typically have a memory, and the memory used is of Secure Digital (SD) type. It has ports for connecting sensors and a computer system. This hardware can be used in the implementation of IoT, smart homes, patient control, and robotics (Jaeho Kim et al., 2018).

#### D. Apache Spark

Big Data is infinitely abundant and cannot be stored or processed in conventional and current memories. Big data are the result of Internet growth, communication tools, social networks, blogs, and computer networks such as IoT. Big data volumes are constantly expanding, and this can be attributed to

an increase in the number of media, smart objects, applications, and content production (M. Chen et al., 2018).

The increased volume of data has made conventional data processing platforms inefficient. Therefore, distributed systems and computations based on them are used to process this volume of data that is continuously increasing. Cloud computing is a distributed metadata computing method whose computational complexity is hidden from users' views. Cloud computing allows a high level of distributed systems to be displayed to users. One way of processing big data is to distribute data and process them on a set of network nodes called clusters (Pai et al., 2018).

Hadoop is one of the new technologies in distributed computing; it is an open-source project from the Apache Foundation for large data processing. It is convenient for big processes. For this purpose, Hadoop performs a distributed process with the help of a set of network clusters and uses map-reduce operations. The map-reduce mechanism has a set of functions for map-reduce operations. In the map functions, the main process is divided into sub-processes. Each sub-process is applied to a part of the distributed data. By mapping functions, the mapping process's intermediate results are aggregated and converted to a final result (Kang et al., 2017). The mechanism and operation of Hadoop architecture map-reduce are shown in Fig. 5:

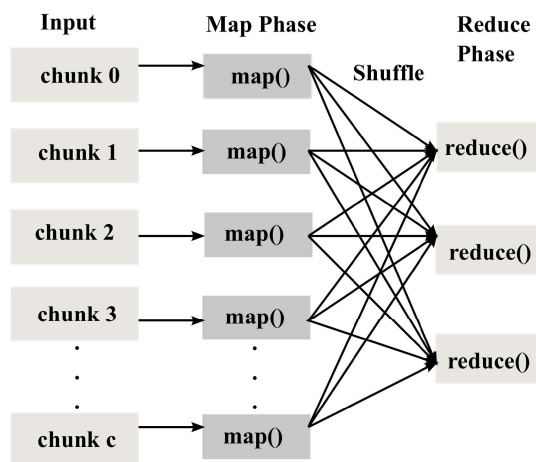


Fig. 5. The map-reduce mechanism in Hadoop distributed architecture (Kang et al., 2017)

As shown in the figure above, distributed data in the Hadoop Distributed File System (HDFS) are separated and divided mapped by the key and value in the mapping stage. The key and combined value results and the results are produced in the reduction stage.

Like Hadoop, Apache Spark is a big data processing platform provided by the Apache Foundation. The mechanism and method of Apache Spark are very similar to the Hadoop model. However, there are differences in their architecture (Islam et al., 2017).

- In Apache Spark architecture, main memory is mostly used, whereas, in Hadoop, secondary memory is more frequently used. Therefore, Apache Spark is faster than Hadoop in processing.

- In Hadoop, there are only map-reduce functions. Apache Spark offers a wide range of functions, APIs and, libraries for big data processing.
- Apache Spark is capable of processing data streams, but Hadoop is not directly capable of doing so.
- Apache Spark has ML libraries, data streams, database processing, and extensive graph analysis, but Hadoop does not have such features.

In Fig. 6, the Apache Spark architecture, and essential components of this ecosystem for big data processing are shown:

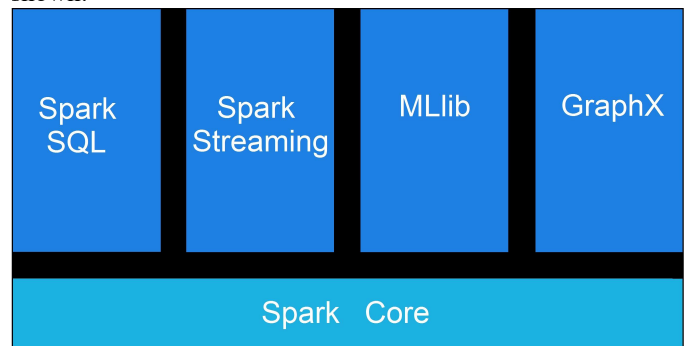


Fig. 6. Apache Spark framework and ecosystem in big data processing (Mavridis & Karatza, 2017)

According to the Apache Spark architecture, the big data processing ecosystem has parts for ML, data stream, graph processing (GraphX), database processing (Spark SQL) and, uses the map-reduce mechanism for these processes.

One of the crucial advantages of the Hadoop and Apache Spark architectures is distributed computing in real-time while making them suitable for practical applications. However, as mentioned before, Apache Spark architecture uses main memory which make it faster (Expósito et al., 2020).

#### E. Diabetes

The large number of diabetic patients worldwide indicates that the disease is a worldwide threat. Diabetes is currently the fourth leading cause of death in developed countries. Diabetics is diagnosed when the amount of blood sugar or glucose reaches a dangerous level, which if too high, can have devastating effects on different organs. One of the vital organs of the body is the pancreas that produces the insulin needed by the cells of the body. This hormone can absorb blood glucose and consume it for cell energy preventing the increase in blood sugar.

There are two significant reasons for abnormally elevated blood sugar levels. In the first case, the pancreas is unable to produce natural and sufficient insulin, which leads to type 2 diabetes. In the second case, the pancreas produces insulin but the body cells cannot absorb insulin for reasons, which can cause type 2 diabetes (Bhanpuri et al., 2018).

Both factors increase blood glucose levels. Diabetes generally has 2 types. In people with type 1 diabetes, the pancreas can produce very little insulin. In a worse state, the pancreas is no longer able to provide the necessary body insulin. In this case, the patient must compensate for the insulin required by daily injections. Type 1 diabetes is mostly seen in children



and adolescents under 20 years of age. However, various reasons for the decrease in insulin production by the pancreas have been discovered, including inheritance, infectious and fungal diseases, lifestyle, environment, and stress. Researchers attribute this to genetic factors. Type 1 diabetes symptoms are varied, including over thirst, hunger, frequent urination, weight loss, fatigue, and impaired vision.

In type 2 diabetes, the pancreas produces a normal amount of insulin hormone but the cells of the body do not absorb it. Decreasing insulin levels in cells affect them by not being able to supply their required glucose well from the blood. They use their cellular resources to supply energy inevitably, which in turn increases blood sugar levels. The proper lifestyle, exercise, and medications of this disease can reduce its effects.

Type 3 diabetes occurs in some cases of pregnancy and usually resolves after pregnancy. More details of diabetes are available in medical reference books (Turksoy et al., 2018).

According to statistics of the WHO, the highest percentage of diabetes is related to type 2 diabetes, which is about 90%. Obesity and inactivity are its leading causes. Most people with this type of diabetes do not show any initial symptoms. After a few years, they become aware of the disease through other diseases. Diabetics can damage the body and its tissues in the long run and affect the body's various organs. Long-term complications of this disease include diseases such as cardiovascular disease, kidney, nerve, eye diseases, and so on.

It should be noted that cardiovascular diseases have the highest mortality in diabetics. The risk of developing diabetes in people increases with increasing weight. There are several clinical trials available to diagnose diabetes, depending on doctors' diagnosis (L. Chen et al., 2018).

### 3. PROPOSED METHOD

In recent years, with the help of the WBAN, medical information and patient-related data can be largely obtained and used for data mining and ML. Patient data collected through medical centers can be of a big data type and may require a long time to be processed.

Therefore, distributed processing such as Apache Spark can reduce the time it takes to find a disease (including diabetes) pattern. To diagnose a disease such as diabetes, the problem needs to be formulated and presented as a knowledge discovery problem. To diagnose diabetes, we can assume that we have a classification problem with two classes. It has a set of features used for learning, as well as steps such as pre-processing, normalization, and data preparation. It has learning techniques to classify the samples into either normal or abnormal classes. The proposed method in this study includes the following steps:

- Collecting data and conducting required pre-processing.
- Importing pre-processed data into Apache Spark environment.
- Conducting map-reduce in a distributed architecture to split tasks and divide data into test and training data.
- Applying learning on Apache Spark distributed machines using training data of diabetes.

- Creating a classification model in two categories of normal and abnormal.
- Testing the learning model distributed by the test data.
- Evaluating the learning model for disease diagnosis by indicators such as runtime and accuracy.

#### A. Framework of the proposed method

This section discusses the framework and steps of the proposed method for diabetes diagnosis by the Apache Spark architecture.

To diagnose diabetic patients, the medical centers collect data for pre-processing, feature selection, and learning, according to Fig. 7:

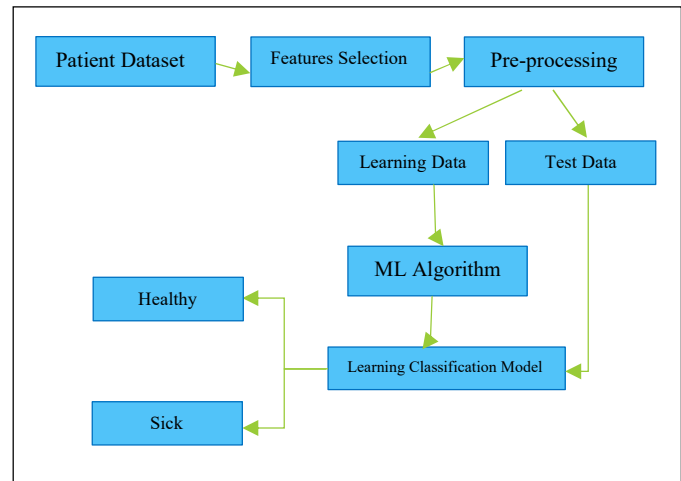


Fig. 7. The framework for diabetes diagnosis using ML mechanism

As shown in the figure above, the diagnosis of diabetic patients requires (1) collection of data on healthy individuals from a data source, (2) extracting important features from such data to enable more accurate learning, and (3) use of a variety of feature selection methods. Apache Spark library has a variety of instructions to choose features from; it will be used in this research. Selecting important features of diabetic patients, the data pre-processing step is performed, where the normalization step is used as the most important part of normalization. Normalization limits the range of any attribute or feature to make learning more accurate. Applying the normalization phase described below, the dataset is divided into two parts of training data and test data, and test data are used for learning and model making.

The learning technique used in the proposed method is a method such as DT and RF with excellent performance and speed. Applying ML, a classification model has been developed to classify individuals into two categories of normal and abnormal. The presented model can be evaluated by test data and analyzed in terms of evaluation indexes such as accuracy and runtime. The above framework is just a learning framework, and the details of the proposed method can be presented as the project and the proposed method in Fig. 8, using the Apache Spark architecture. According to the figure, in the proposed method, the dataset of diabetic patients is

converted into Resilient Distributed Dataset (RDD) format. This format is used for storage. RDD format is a way of storing in an Apache Spark distributed memory architecture or HDFS.

In the proposed method, there are several clusters of computing nodes that are divided into two primary and secondary classes such in a way that the main cluster has the task of distributing the computational burden, including ML over other Spark nodes, and is responsible for managing the sub-clusters. Distributing data related to patients or healthy individuals in the Apache Spark Distribution System, ML techniques can be performed on the data available in the Spark Distributed File System with the help of mapping operation.

At this step, each of the secondary nodes performs part of the learning on the data associated with the node, in better words, the distributed learning technique is applied by mapping functions to the data loaded into the node's main memory.

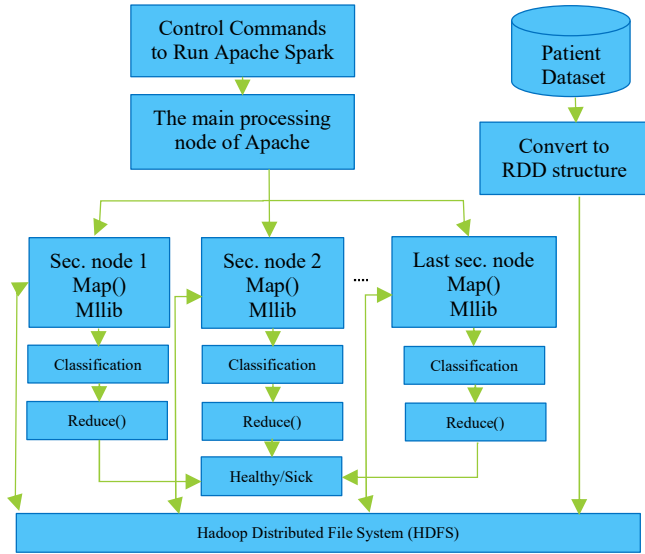


Fig. 8. A suggested framework for diabetes diagnosis

Unlike Hadoop, Apache Spark does not require the use of auxiliary tools such as mammoths to do ML, but it has a rich ML library called Mllib that contains ML commands and techniques. These techniques are applied to the Spark Distributed File System with the help of map-reduce operations.

The data from the medical samples are converted to Spark Distributed System format or RDD. Each node loads part of this data into its memory, and the ML technique is applied in parallel to each node. The master node also manages each secondary node during learning; this management can include resource management such as memory. Each of the secondary nodes apply a classification model to a portion of the data. Then, the classification output of the secondary nodes can be aggregated and reduced by the reduction technique.

### B. Normalization

Information about healthy or sick individuals has a set of features, each of which has a specific range of numbers defining the attribute. attribute's value, if restricted to a small range,

accelerate learning by data mining and ML methods and increase data accuracy.

Normalization is an important process in learning and discovering knowledge, and is referred to as a process that maps the values of an attribute from a larger map interval to a smaller one. To normalize the attributes associated with healthy or sick individuals, equation (1) can be used in the interval  $[a, b]$ :

$$N = \frac{n - \min}{\max - \min} (b - a) + b \quad (1)$$

In this equation,  $N$  is the normalized value of an attribute,  $n$  is the unnormalized value of an attribute,  $\min$  and  $\max$  are the minimum or maximum of an attribute, respectively, and  $a$  and  $b$  are the lower and upper limits of normalization, respectively. The values of each attribute can be normalized in the interval  $[0,1]$ .

### C. Feature selection

One of the most important steps in ML in the diagnosis of diabetes is the optimal selection of attributes; Apache Spark has presented itself as a good choice for ML and classification applications. In the proposed method, for learning and creating a classification model, the values of attributes must be made numerical as the first step. Then, the learning technique must be applied by the Mllib library commands. For this purpose, the HashingTF class in Scala or Python is used to develop the proposed method. This package allows creating an object of the desired class to select the important attributes and applying learning to them. An example of the commands is illustrated below:

```
import org.apache.spark.mllib.feature.HashingTF
val hash = new HashingTF()
val features=hash(SizeFeatures)
```

In the command on the first line, the feature selection class is added to the Scala language. The second line shows that an object of this class is created. Finally, in the third command, many important attributes are selected for learning. Scala selects the most important attributes for learning with the help of this function. The selected attributes must be applied to the data used as the data stream, so it is necessary to define a mapping between the input data and the attribute used in the dataset. Thus, according to the commands below, the *LabeledPoint* class package is added. Then, using the second command, the selected attributes are applied to the input or data to apply the important data to medical examples for learning.

```
import org.apache.spark.mllib.LabeledPoint
val data =features.map(features =>
LabeledPoint(1,features))
```

### D. Proposed flowchart

In Fig. 9, a flowchart of the proposed method steps in the diagnosis of diabetes is presented using the Apache Spark framework and architecture. According to the proposed flowchart (a) the data are first imported to the Apache Spark distributed file system via a data stream interface, (b) feature

selection is performed on the data, followed by medical examples and data normalization, and, (d) ML method such as decision tree is applied.

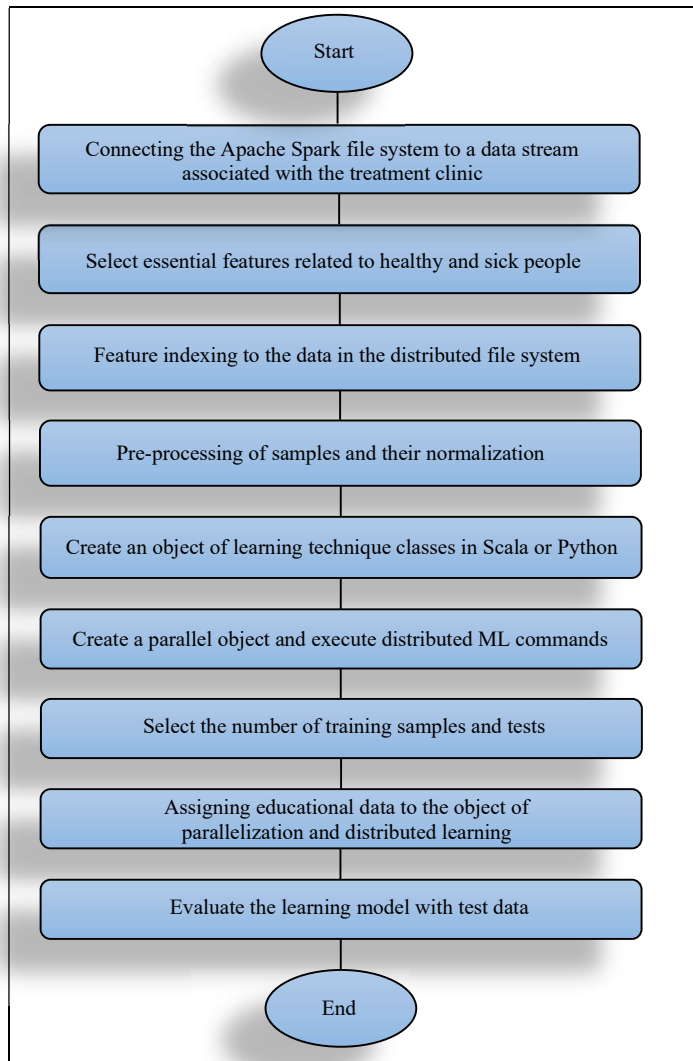


Fig. 9. Flowchart of the proposed method in the diagnosis of diabetes

As shown, the first step of data entry is done by a WBAN or a dataset that can be converted into an appropriate Apache Spark format and uploaded to its distributed file system. Then important features are selected with library commands of Spark. These selected attributes will eventually be linked to the data and associated attributes. In the proposed method, data normalization is performed after the feature selection to make the learning more accurate and less sensitive to the data attribute values. After the normalization step, we create an object related to ML such as the Decision Tree (DT) technique or Random Forest (RF) or Support Vector Machine (SVM).

To run in parallel and in a distributed way on the Spark nodes, this object needs to run in the context of a parallel Spark object, which is created at this stage. The role of the parallel object is to implement and apply ML techniques on a set of Spark computational clusters. The learning object implemented in the context of the Spark parallelization object needs to use

part of the data as training data in order to create a classification model so that healthy samples can be isolated from the sick patients. At the end of the proposed flowchart, there is an evaluation section, which can calculate the accuracy and timing of the proposed method in the diagnosis of diabetic patients using test data. In this study, the proposed method presented for the diagnosis of diabetes in Apache Spark has the following advantages:

- Due to the simultaneous processing of Apache Spark, nodes or clusters diagnosis is reduced.
- The Apache Spark architecture is more advanced than the Hadoop. So, the proposed method used in this platform has the benefits of Spark like the use of main memory instead of sub-memory.
- Disease diagnosis using distributed processing techniques such as Apache Spark is performed in real-time and can be used to diagnose the disease quickly.

Although distributed systems such as Apache Spark are fast and accurate for disease diagnosis, these methods also have their limitations, such as hardware and high complexity, which make their development challenging.

#### 4. IMPLEMENTATION AND EVALUATION

In this section, the implementation and simulation environment for the diagnosis of diabetes in the Apache Spark environment is evaluated using an appropriate dataset and the results are reported.

##### A. Data collection

The study used data from a database of 130 U.S. hospitals on diabetes that has 55 different features. This dataset contains various attributes used to diagnose diabetes (Hu et al., 2017). For example, if the gender is female, the number of pregnancies is an important indicator in the diagnosis of diabetes. The Oral Glucose Tolerance Test (OGTT) is a clinical trial in the diagnosis of diabetes. The fasting blood sample is taken from a person, then 75 grams of oral glucose is given to the person, and the blood sample is retaken after two hours.

The blood glucose in the second sample in the deciliter scale is used as a criterion for measuring diabetes in this dataset. Triceps skinfold thickness is an essential indicator in measuring diabetes, expressed in millimeters. Blood insulin value two hours after breakfast is an important indicator of measuring diabetes. Body Mass Index ( $BMI = \frac{kg}{m^2}$ ) where kg is a person's weight in kilograms and  $m^2$  is their height in meters squared, can be used to measure diabetes. Diabetes pedigree function in the family and the relatives, calculated as a number, is one of the most criteria in measuring a person's disease.

##### B. Implementation environment

Apache Spark requires considerable memory to execute, so it is recommended to use systems with at least 4 GB of memory because unlike Hadoop, in Apache Spark, the data which are the patient-related samples here can be loaded into the main memory using less secondary memory which leads to an

increase in Apache Spark's performance over Hadoop.

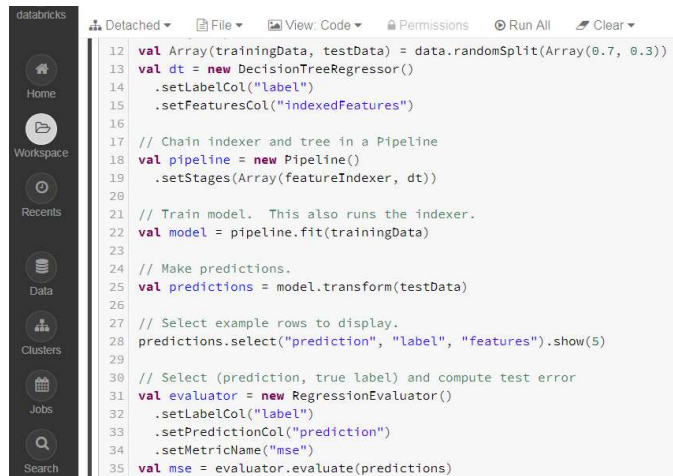
### C. Analysis

Classification of the proposed method requires that the objective function of the problem be accurately determined to provide a more accurate classification of the desired problem. Classification usually uses the criterion of the Mean Square Error (MSE) or the Root Means Square Error (RMSE) as the objective or cost function.

In other words, these two criteria, which are considered equivalent to each other, are used to measure the quality of the classification. If these values are selected to be more minimum, the classification will be more accurate. Ideally, these values are approaching zero, indicating the minimum classification error and effectiveness of the corresponding algorithm. The following equation (2) shows the MSE for diabetic patients' classification:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2)$$

In this equation,  $y_i$  is the label value or actual class number of the  $i$ th sample or data,  $\hat{y}_i$  is the predicted value of the label or class number of the  $i$ th sample or data, and  $N$  is the number of samples or sick or healthy individuals. This paper utilizes Scala to implement DT, RF, and SVM techniques in the Spark environment. An example of the DT technique implementation is shown in Fig. 10:



```

12 val Array(trainingData, testData) = data.randomSplit(Array(0.7, 0.3))
13 val dt = new DecisionTreeRegressor()
14   .setLabelCol("label")
15   .setFeaturesCol("indexedFeatures")
16
17 // Chain indexer and tree in a Pipeline
18 val pipeline = new Pipeline()
19   .setStages(Array(featureIndexer, dt))
20
21 // Train model. This also runs the indexer.
22 val model = pipeline.fit(trainingData)
23
24 // Make predictions.
25 val predictions = model.transform(testData)
26
27 // Select example rows to display.
28 predictions.select("prediction", "label", "features").show(5)
29
30 // Select (prediction, true label) and compute test error
31 val evaluator = new RegressionEvaluator()
32   .setLabelCol("label")
33   .setPredictionCol("prediction")
34   .setMetricName("mse")
35 val mse = evaluator.evaluate(predictions)

```

Fig. 10. Implementation of Spark decision tree technique in Scala

An example of the output of the DT algorithm in Spark's operational environment for diabetes diagnosis is shown in Fig. 11:

```

+-----+-----+-----+
| prediction|label|      features|
+-----+-----+-----+
|[0.00910706352732119]| 0.0|[30,[0,1,2,3,4,5,...]|
|[0.00910706352732119]| 0.0|[30,[0,1,2,3,4,5,...]|
|[0.00910706352732119]| 0.0|[30,[0,1,2,3,4,5,...]|
|[0.00910706352732119]| 0.0|[30,[0,1,2,3,4,5,...]|
|[0.00910706352732119]| 0.0|[30,[0,1,2,3,4,5,...]|
+-----+-----+-----+
only showing top 5 rows

Mean Squared Error or MSE= 0.10141898154346475
Treemodel:
DecisionTreeRegressionModel (uid=dtr_ce189ae98fb) of depth 5 with 57 nodes
  If (feature 13 <= 0.0)
    If (feature 1 <= 127.5)
      If (feature 7 <= 28.5)
        If (feature 5 <= 30.75)
          If (feature 0 <= 7.5)
            Predict: 0.00910706352732119
          Else (feature 0 > 7.5)

```

Fig. 11. The output of the decision tree technique in Spark

The above output shows the different levels of the tree and its properties. In this experiment, the mean error is assumed to be 0.101. Indeed, in each experiment, the MSE value can be slightly different. To obtain the required results, the desired test can be repeated a specified number of times to express the mean of this index. The tree structure of the decision tree can be graphically represented by the display command in Scala, part of which is shown in Fig. 12:

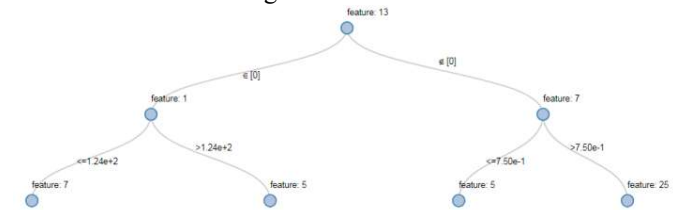


Fig. 12. The output of the decision tree technique in Spark as a tree

Other learning techniques can be placed in Scala as the above procedure and show their output in the diagnosis of diabetes. To obtain the mean results in the evaluations in this paper, we consider the number of trials for each technique 32 times and evaluate their MSE as a criterion. Evaluation and comparison of the MSE of running the algorithms of DT, RF, and SVM in Spark distributed and non-distributed modes in WEKA are shown in Table 1. They are also shown graphically in Fig. 13 for a better understanding of the diagnosis error rate in each of these methods.

**Table 1**  
Comparison of mse index in (spark) distributed and non-distributed modes

Methods	Error in WEKA	Error in Spark
DT	0.199	0.187
RF	0.179	0.165
SVM	0.226	0.221

Based on the experiments conducted, it can be found that the mean error in the non-distributed computing mode for the diagnosis of diabetes in the three techniques of DT, RF, and SVM are 0.199, 0.179, and 0.226, respectively, while in the undistributed mode, these values are 0.187, 0.165, and 0.221, respectively, which are fewer values.

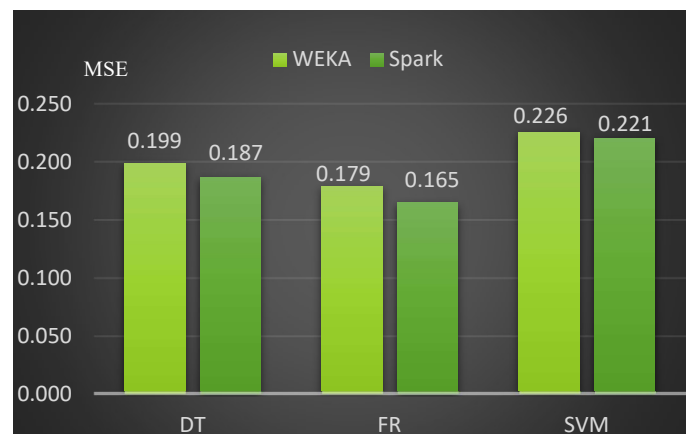


Fig. 13. Comparison of MSE Index Distributed/ Non-Distributed Modes in Spark



A comparison of three techniques of DT, RF, and SVM in the diagnosis of diabetes shows that the random forest method is more effective in diagnosing diabetes. A comparison of the runtime of the algorithms in WEKA and Spark is shown in Fig. 14, to compare the effect of running algorithms and their time in distributed Spark mode and normal mode in WEKA. In this comparison, it is assumed that clusters with 6 GB memory are used. Additionally, for the calculation of the runtime, each of the techniques of DT, RF, and SVM in the diagnosis of diabetes are executed 32 times. Here, we calculate their average runtime in distributed and non-distributed modes.

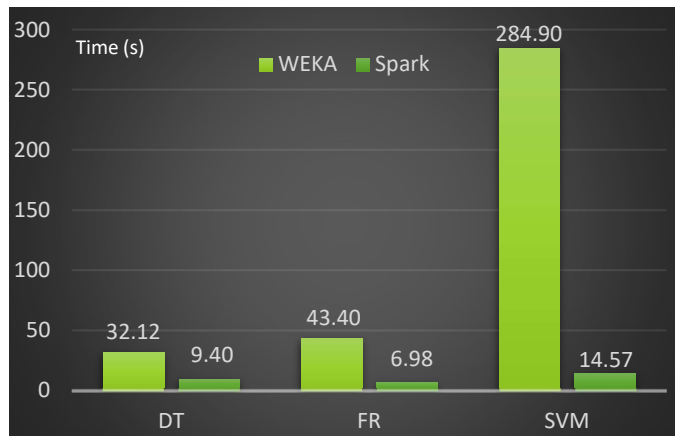


Fig. 14. Comparison of runtime in (Spark) distributed and non-distributed modes in the diagnosis of diabetes

According to the above graph, data mining methods, especially DT and RF, have much less time in Spark than non-spark. On the other hand, the random forest technique has less runtime (about 6.98 seconds) to detect diabetes in Spark. The important advantage of the proposed method is providing a platform and pathway for big data processing in modern distributed systems such as Apache Spark in the diagnosis of diabetes in a short time and is appropriate accuracy. However, using big data processing methods in distributed systems such as Apache Spark needs strong hardware as well as high-level programming knowledge. These features make it difficult for the public to use these frameworks.

## 5. CONCLUSION AND FUTURE WORK

In this study, an information processing framework in the Spark environment has been developed to diagnose diabetes. The proposed method uses ML and data mining techniques such as DT, RF, and SVM in the platform of the Apache Spark environment to analyze large-scale medical samples to diagnose diabetes. In this study, these techniques were implemented in non-distributed and distributed environments such as WEKA and Apache Spark system. Experimental results show that the implementation of learning techniques such as decision DT, RF, and SVM in Apache Spark reduces their runtime, such that the runtime in a non-distributed system and Spark is equal to 43.40 and 6.98 seconds, respectively. Our study results show that the Apache Spark framework can

increase the accuracy of learning techniques in diagnosing the disease while reducing the runtime. One of our future research goals is to integrate WBAN with Apache Spark technology to diagnose patient's status in the hospital.

## REFERENCES

- Aich, S., Younga, K., Hui, K. L., Al-Absi, A. A., & Sain, M. (2018). A nonlinear decision tree based classification approach to predict the Parkinson's disease using different feature sets of voice data. *International Conference on Advanced Communication Technology, ICACT, 2018-Febru*, 638–642. <https://doi.org/10.23919/ICACT.2018.8323864>
- Al-Janabi, S., Al-Shourbaji, I., Shojafar, M., & Shamshirband, S. (2017). Survey of main challenges (security and privacy) in wireless body area networks for healthcare applications. *Egyptian Informatics Journal*, 18(2), 113–122. <https://doi.org/10.1016/j.eij.2016.11.001>
- Al-Turjman, F., & Deebak, B. D. (2021). Seamless Authentication: For IoT-Big Data Technologies in Smart Industrial Application Systems. *IEEE Transactions on Industrial Informatics*, 17(4), 2919–2927. <https://doi.org/10.1109/TII.2020.2990741>
- Baker, S. B., Xiang, W., & Atkinson, I. (2017). Internet of Things for Smart Healthcare: Technologies, Challenges, and Opportunities. *IEEE Access*, 5, 26521–26544. <https://doi.org/10.1109/ACCESS.2017.2775180>
- Barale, M. S., & Shirke, D. T. (2016). Cascaded Modeling for PIMA Indian Diabetes Data. *International Journal of Computer Applications*, 139(11), 1–4. <https://doi.org/10.5120/ijca2016909426>
- Bequette, B. W., Cameron, F., Buckingham, B. A., Maahs, D. M., & Lum, J. (2018). Overnight Hypoglycemia and Hyperglycemia Mitigation for Individuals with Type 1 Diabetes: How Risks Can Be Reduced. *IEEE Control Systems*, 38(1), 125–134. <https://doi.org/10.1109/MCS.2017.2767119>
- Bhanpuri, N. H., Hallberg, S. J., Williams, P. T., McKenzie, A. L., Ballard, K. D., Campbell, W. W., McCarter, J. P., Phinney, S. D., & Volek, J. S. (2018). Cardiovascular disease risk factor responses to a type 2 diabetes care model including nutritional ketosis induced by sustained carbohydrate restriction at 1 year: An open label, non-randomized, controlled study. *Cardiovascular Diabetology*, 17(1), 1–16. <https://doi.org/10.1186/s12933-018-0698-8>
- Chen, L., Lu, D., Pirbhulal, S., Sodhro, A. H., Chen, Z., Huang, G., & Wu, H. (2018). Development of knowledge-based ontology framework for diabetes patients in medical applications. *International Conference on Biological Information and Biomedical Engineering, BIBE 2018*, 448–451.
- Chen, M., Yang, J., Zhou, J., Hao, Y., Zhang, J., & Youn, C. H. (2018). 5G-Smart Diabetes: Toward Personalized Diabetes Diagnosis with Healthcare Big Data Clouds. *IEEE Communications Magazine*, 56(4), 16–23. <https://doi.org/10.1109/MCOM.2018.1700788>
- Das, H., Naik, B., & Behera, H. S. (2018). Classification of diabetes mellitus disease (DMD): A data mining (DM) approach. In *Advances in Intelligent Systems and Computing* (Vol. 710, pp. 539–549). Springer. [https://doi.org/10.1007/978-981-10-7871-2\\_52](https://doi.org/10.1007/978-981-10-7871-2_52)
- Dutt, A., Ismail, M. A., & Herawan, T. (2017). A Systematic Review on Educational Data Mining. *IEEE Access*, 5, 15991–16005. <https://doi.org/10.1109/ACCESS.2017.2654247>
- Expósito, R. R., González-Domínguez, J., & Touriño, J. (2020). SMusket: Spark-based DNA error correction on distributed-memory systems. *Future Generation Computer Systems*, 111, 698–713. <https://doi.org/10.1016/j.future.2019.10.038>
- Fernandes, D., Ferreira, A. G., Abrishambaf, R., Mendes, J., & Cabral, J. (2018). Survey and taxonomy of transmissions power control mechanisms for wireless body area networks. *IEEE Communications Surveys and Tutorials*, 20(2), 1292–1328. <https://doi.org/10.1109/COMST.2017.2782666>
- He, D., Zeadally, S., & Wu, L. (2018). Certificateless Public Auditing Scheme for Cloud-Assisted Wireless Body Area Networks. *IEEE Systems Journal*, 12(1), 64–73. <https://doi.org/10.1109/JSYST.2015.2428620>
- Hu, J., Chen, X., Wang, Y., Huang, Y., & Su, X. (2017). Cloud-assisted home health monitoring system. *Proceedings - 16th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2017*, 899–903. <https://doi.org/10.1109/ICIS.2017.7960120>
- Humayun, A. I., Tauhiduzzaman Khan, M., Ghaffarzadegan, S., Feng, Z., &

- Hasan, T. (2018). An ensemble of transfer, semi-supervised and supervised learning methods for pathological heart sound classification. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, 2018-Septe*, 127–131. <https://doi.org/10.21437/Interspeech.2018-2413>
- Islam, M. T., Karunasekera, S., & Buyya, R. (2017). dSpark: Deadline-based resource allocation for big data applications in apache spark. *Proceedings - 13th IEEE International Conference on eScience, ESScience 2017*, 89–98. <https://doi.org/10.1109/eScience.2017.21>
- Kang, K. D., Chen, L., Yi, H., Wang, B., & Sha, M. (2017). Real-time information derivation from big sensor data via edge computing. *Big Data and Cognitive Computing, 1*(1), 1–24. <https://doi.org/10.3390/bdcc1010005>
- Kim, Jaeho, Choi, S. C., Yun, J., & Lee, J. W. (2018). Towards the oneM2M standards for building IoT ecosystem: Analysis, implementation and lessons. *Peer-to-Peer Networking and Applications, 11*(1), 139–151. <https://doi.org/10.1007/s12083-016-0505-9>
- Kim, Jaehoon, Oh, J., & Heo, T.-Y. (2021). Acoustic Scene Classification and Visualization of Beehive Sounds Using Machine Learning Algorithms and Grad-CAM. *Mathematical Problems in Engineering, 2021*, 1–13. <https://doi.org/10.1155/2021/5594498>
- Komal Kumar, N., Vigneswari, D., Vamsi Krishna, M., & Phanindra Reddy, G. V. (2019). An optimized random forest classifier for diabetes mellitus. In *Advances in Intelligent Systems and Computing* (Vol. 813, pp. 765–773). Springer. [https://doi.org/10.1007/978-981-13-1498-8\\_67](https://doi.org/10.1007/978-981-13-1498-8_67)
- Mavridis, I., & Karatzas, H. (2017). Performance evaluation of cloud-based log file analysis with Apache Hadoop and Apache Spark. *Journal of Systems and Software, 125*, 133–151. <https://doi.org/10.1016/j.jss.2016.11.037>
- Moreira, M. W. L., Rodrigues, J. J. P. C., Marcondes, G. A. B., Neto, A. J. V., Kumar, N., & Diez, I. D. L. T. (2018). A Preterm Birth Risk Prediction System for Mobile Health Applications Based on the Support Vector Machine Algorithm. *IEEE International Conference on Communications, 2018-May*, 1–5. <https://doi.org/10.1109/ICC.2018.8422616>
- Pai, P. P., Sanki, P. K., Sahoo, S. K., De, A., Bhattacharya, S., & Banerjee, S. (2018). Cloud Computing-Based Non-Invasive Glucose Monitoring for Diabetic Care. *IEEE Transactions on Circuits and Systems I: Regular Papers, 65*(2), 663–676. <https://doi.org/10.1109/TCSI.2017.2724012>
- Ramirez-Gallego, S., Mourino-Talin, H., Martinez-Rego, D., Bolon-Canedo, V., Benitez, J. M., Alonso-Betanzos, A., & Herrera, F. (2018). An Information Theory-Based Feature Selection Framework for Big Data under Apache Spark. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 48*(9), 1441–1453. <https://doi.org/10.1109/TSMC.2017.2670926>
- Ruano, M. G., Almeida, G. P., Palma, F., Raposo, J. F., & Ribeiro, R. T. (2018). Reliability of medical databases for the use of real word data and data mining techniques for cardiovascular diseases progression in diabetic patients. *2018 Global Medical Engineering Physics Exchanges/Pan American Health Care Exchanges, GMEPE/PAHCE 2018*, 1–6. <https://doi.org/10.1109/GMEPE-PAHCE.2018.8400769>
- Sakumura, Y., Koyama, Y., Tokutake, H., Hida, T., Sato, K., Itoh, T., Akamatsu, T., & Shin, W. (2017). Diagnosis by volatile organic compounds in exhaled breath from lung cancer patients using support vector machine algorithm. *Sensors (Switzerland), 17*(2), 287. <https://doi.org/10.3390/s17020287>
- Sarabia-Jacome, D., Belsa, A., Palau, C. E., & Esteve, M. (2018). Exploiting IoT data and smart city services for chronic obstructive pulmonary diseases risk factors monitoring. *Proceedings - 2018 IEEE International Conference on Cloud Engineering, IC2E 2018*, 351–356. <https://doi.org/10.1109/IC2E.2018.00060>
- Shickel, B., Tighe, P. J., Bihorac, A., & Rashidi, P. (2018). Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis. *IEEE Journal of Biomedical and Health Informatics, 22*(5), 1589–1604. <https://doi.org/10.1109/JBHI.2017.2767063>
- Tan, J., Xiong, T., Miao, H., Sun, R., & Wu, M. (2018). A case study of medical big data processing: Data mining for the hyperuricemia. *2018 3rd IEEE International Conference on Cloud Computing and Big Data Analysis, ICCCBDA 2018*, 196–201. <https://doi.org/10.1109/ICCCBDA.2018.8386511>
- Taralunga, D. D., & Florea, B. C. (2021). A blockchain-enabled framework for mhealth systems. *Sensors, 21*(8), 2828. <https://doi.org/10.3390/s21082828>
- Turksay, K., Littlejohn, E., & Cinar, A. (2018). Multimodule, Multivariable Artificial Pancreas for Patients with Type 1 Diabetes: Regulating Glucose Concentration under Challenging Conditions. *IEEE Control Systems, 38*(1), 105–124. <https://doi.org/10.1109/MCS.2017.2766326>
- Wu, H., Yang, S., Huang, Z., He, J., & Wang, X. (2018). Type 2 diabetes mellitus prediction model based on data mining. *Informatics in Medicine Unlocked, 10*, 100–107. <https://doi.org/10.1016/j.imu.2017.12.006>
- Xu, Z., Shen, D., Nie, T., & Kou, Y. (2020). A hybrid sampling algorithm combining M-SMOTE and ENN based on Random forest for medical imbalanced data. *Journal of Biomedical Informatics, 107*, 103465. <https://doi.org/10.1016/j.jbi.2020.103465>
- Zarkogianni, K., Athanasiou, M., & Thanopoulou, A. C. (2018). Comparison of Machine Learning Approaches Toward Assessing the Risk of Developing Cardiovascular Disease as a Long-Term Diabetes Complication. *IEEE Journal of Biomedical and Health Informatics, 22*(5), 1637–1647. <https://doi.org/10.1109/JBHI.2017.2765639>