Kiran Tangod* and Gururaj Kulkarni

Secure Communication through MultiAgent System-Based Diabetes Diagnosing and Classification

https://doi.org/10.1515/jisys-2017-0353
Received July 18, 2017; previously published online June 29, 2018.

Abstract: The main objective of the research is to provide a multi-agent data mining system for diagnosing diabetes. Here, we use multi-agents for diagnosing diabetes such as user agent, connection agent, updation agent, and security agent, in which each agent performs their own task under the coordination of the connection agent. For secure communication, the user symptoms are encrypted with the help of Elliptic Curve Cryptography and Optimal Advanced Encryption Standard. In Optimal Advanced Encryption Standard algorithm, the key values are optimally selected by means of differential evaluation algorithm. After receiving the encrypted data, the suggested method needs to find the diabetes level of the user through multiple kernel support vector machine algorithm. Based on that, the agent prescribes the drugs for the corresponding user. The performance of the proposed technique is evaluated by classification accuracy, sensitivity, specificity, precision, recall, execution time and memory value. The proposed method will be implemented in JAVA platform.

Keywords: Multi-agent diabetes, advanced encryption standard, elliptic curve cryptography, differential evaluation, multiple kernel support vector machine algorithm.

1 Introduction

Healthcare is currently a major problem in many developing countries [14]. One such healthcare issue is identified with diabetes. Diabetes is all inclusive and expanding at a disturbing rate. It is a perplexing condition that influences all aspects of a person's life and requires excellent consideration. It occurs due to increase in glucose level [8, 18]. According to the International Diabetes Federation, more than 140 million individuals worldwide, and 50 million individuals in India are experiencing diabetes. Approximately 347 million individuals worldwide have diabetes, and according to the World Health Organization, that diabetes mortality will increase two-fold between 2005 and 2030. The death rate before age of 35 years is 65%, basically because to focal respiratory and/or renal failure [12]. Diabetes is the most widely recognized endocrine issue in which the body does not create or appropriately utilize insulin [1, 23]. One of the main typical microscale vascular intricacies of diabetes is diabetic retinopathy and non-diabetic renal ailments, which affect 93 million individuals and is the main cause of visual deficiency in middle-aged population worldwide [24].

In India, most human service conveyance frameworks depend on continuous manual recording, including ophthalmology. In ophthalmology, achieving a fact-driven analysis is never a simple occupation for a clinician [6]. There are different examinations to be done which have distinctive parameters that have to be analyzed individually and constructed to reach to a definite analysis [17]. As an underlying issue in the area of healthcare consideration and diagnosis, we focused on online medicinal administration framework. The soft computing method gives adaptable data-preparing capacity to take care of real-life questionable circumstances in

***Corresponding author: Kiran Tangod,** Department of Information Science and Engineering, Gogte Institute of Technology, Belagavi, Karnataka, India, e-mail: kirantangod1178@gmail.com
Gururaj Kulkarni: Department of Electrical and Electronics Engineering, Jain College of Engineering, Belagavi, Karnataka, India

correlation with ordinary hard processing. The Internet is widely used. Medical diagnosis is known to be subjective for several reasons: first, it relies on the physician making the diagnosis. Second, and most essentially, the measure of information that should be analyzed to make a good prediction is usually huge and at times uncontrollable [19]. In addition, treatment and creation of a medical diagnosis system composed of operators using JADE constitute a multi-agent system [7, 16]. The agent-based framework, which is a piece of artificial intelligence, has turned into a developing range that controls and executes straightforward or complex issues.

An agent is a PC program intended to carry out a task on behalf of its client or proprietor. An agent must have the following characteristics: autonomous, pro-action, reactivity, communication, cooperation, negotiation, learning, and so on [2]. In an agreeable multi-agent system, agents can partition an issue to a few sub-issues and impart their insight to each other. Result sharing is one the most vital capacities of a multi-agent system, where every agent shares its nearby result acquired in light of its neighborhood information with different agents. Thus, it gives more precise results on diagnosis [22]. However, there are various medical diagnosis systems such as multi-agent–based clinical diagnoses system that takes care of the early stage of the patient, but most of the systems are not capable of diagnosing and monitoring patient online. Hence, there is need to develop a multi-agent-based system that will be able to diagnose and monitor a patient with chronic disease such as diabetes and update the doctor with the patient's progress [14]. Further, for diagnosis, a span expert system is utilized. It is separated into two sections: the deduction motor, which is settled and free of the expert system, and the learning base, which has one variable [10]. Using all these methods, diabetes can be easily diagnosed and further treatment can be provided for its reduction. The performance of all these methods in the diagnosis of diabetes is extremely high.

2 Literature Survey

In a case-controls study by Zhang et al. [20], a relationship was found between leptin receptor (*LEPR*) gene polymorphism and type 2 diabetes mellitus (T2DM); however, the outcomes had not been generally predictable among different people. The meta-analysis was intended to survey a more exact relationship between *LEPR* polymorphism and T2DM. Eight electronic databases were checked for Chinese and English articles published between 2000 and 2015 that studied the association between *LEPR* polymorphism and T2DM. Pooled odds ratios (OR) with a 95% confidence interval (CI) were computed in terms of allele contrast, latent, and prevailing and added substance genetic models to survey that association.

Le et al. [11] analyzed how socioeconomic variables were connected with prevalence and self-administration of diabetes among ethnic minority groups in the rural Yunnan area, which had the most number of ethnic minority groups per region in southwest China. A cross-sectional overview was completed in 2014 in a rural southwest population comprising 5532 subjects older than 35 years who provided consent. Data about members' demographic characteristic and diagnosis of diabetes, therapeutic measures, and self-management practices were acquired utilizing a standard poll. Fasting glucose levels were recorded for every person. A socioeconomic position (SEP) list was built utilizing principal component analysis.

To examine treatment and accomplishment of glycemic in patients' mind focuses T2DM treated with basal insulin in a certifiable setting and to decide doctors' convictions and works on regard to these patients. The study of Dalal et al. [5] had two parts: a review of a US claims database of patient and treatment information and a study of doctors' convictions and practices. It showed that many patients on insulin-based treatment did not reach their glycemic objectives. Further training of clinicians might enhance the insulin strengthening rates and increase the number of patients achieving glycemic targets.

Willis et al. [25] led a study that evaluated the attainability of a faith center-based method for screening and referring high-risk persons to train in glycemic uptake and decreasing diabetes risk. It focuses skillful screening and early intercession procedures for individuals at risk due to diabetes and cardiovascular conditions in nearby confidence. The screening procedure is used to compare diabetes risk evaluation apparatus and a close patient test for HbA_{1c}. Those observed to be at high risk of diabetes (HbA_{1c} 6–6.4%/42–46 mmol/mol)

were offered a "Mobile Away from Diabetes" educational intervention, which aimed at increasing exercise levels and diminishing diabetes risk.

Interleukin 6 (IL-6), a fiery cytokine, is viewed as a hopeful gene conceivably required in nephropathy in diabetes. A study by Chang et al. [3] investigated whether IL-6 polymorphisms anticipated the progression of nephropathy in a cohort of Chinese patients with T2DM. An aggregate of 568 T2DM patients with normoalbuminuria was followed up for a mean of 5.3 ± 1.5 years. Urinary albumin-to-creatinine ratio (ACR) of 30 mg/g in two successive urine tests were characterized as progression to diabetic nephropathy (n = 143). Five polymorphisms of the *IL-6* gene, rs1800795, rs1800796, rs1524107, rs2069837, and rs2069840, were genotyped. Cox corresponding peril models were utilized to gauge the hazard ratio (HR) and 95% CI of progression to diabetic nephropathy under various genetic designs.

Epidemiological confirmation recommends that adipokines might be connected with the onset of T2DM, but this has not been proven to date. Neville et al. [13] investigated the relationship among adiponectin and leptin and the resulting diagnosis of T2DM in a UK population-based cohort of non-diabetic middle-aged men. Gauge serum levels of leptin and adiponectin were measured in 1839 for non-diabetic men aged 50–60 years who were joined the population-based PRIME study. Over a mean follow-up of 14.7 years, new instances of T2DM were resolved from self-reported clinical data with ensuing acceptance by general specialists.

Chen et al. [4] assessed the viability of sitagliptin in 1874 Taiwanese T2DM subjects with various baseline BMIs in a single-focus, healing center-based review diagram audit. Subjects were characterized into subgroups based on their baseline using Taiwan's national weight classification: typical (BMI <24 kg/m^2) (n = 504), overweight (BMI: 24–27 kg/m^2) (n = 615), and large (BMI P 27 kg/m^2) (n = 755). Changes in HbA_{1c} and weight were assessed over a 12-month therapeutic duration.

3 Motivation of the Proposed Technique

In the existing technique, multi-agent-based medical diagnosis and classification with the aid of hybrid firefly-based neural network were proposed [21]. Here, the user agent collects the user symptoms and then sends the request message to the connection agent. After receiving the request, the connection agent makes an initial connection with the updation agent. For secure communication between the user and the updation agent, the existing technique uses cryptography. Here, the Advanced Encryption Standard (AES) algorithm is used for secure communication. Finally, the updation agent prescribes the drugs for the corresponding user. So that existing technique uses the hybrid firefly neural network algorithm is used to classify diabetes and the result shows the secure communication and also maximum classification accuracy but also it has some disadvantages, the communication between connection agent and updation agent is secure in existing technique but if any attacks or miss communication happen between user agent and connection agent it will affect the entire communication to overcome this problem the proposed technique use more secure way of communication between user agent, connection agent and updation agent. The proposed technique utilizes Elliptic Curve Cryptography (ECC) and Optimal Advanced Encryption Standard (OAES) for dual encryption and MKSVM for diabetes prediction.

4 Proposed Methodology

A multi-agent system consists of several agents interacting reciprocally with other agents both within themselves and with their surrounding environment. For secure communication, the user symptoms are encrypted through ECC and OAES algorithm. In the AES algorithm, the key value is optimally selected by means of differential evaluation (DE) algorithm. Finally, the encrypted data are fed to the updation agent for diabetes diagnosis. Thus, our proposed technique uses the multiple kernel support vector machine algorithm (MKSVM) to classify the diabetes level based on the drug prescribed by the updation agent to the corresponding user. The overall flow diagram of the proposed technique is shown in Figure 1.

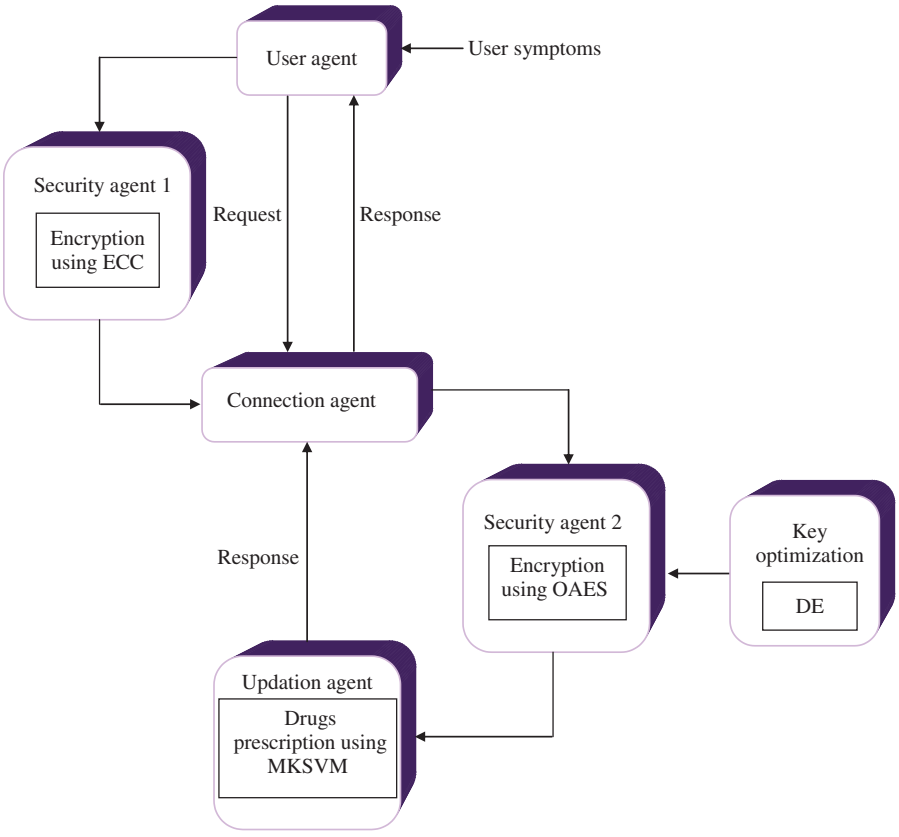

Figure 1: Overall Flow Diagram of the Proposed Technique.

The proposed method uses multi-agents for diabetes diagnosis namely, user agent, connection agent, updation agent and security agent. In our proposed technique, two types of security agent are used for secure communication, namely security agent 1 and security agent 2. Between the user agent and the connection agent, security agent 1 is used with the help of ECC. Security agent 2 is used between the connection agent and the updation agent by means of OAES. The details of the functionality of each agent are described as follows:

User agent: A user agent collects the user symptoms and sends the user symptoms by sending a request message to the connection agent. It waits to receive the response from the connection agent.

Connection agent: A connection agent is responsible for connecting the user agent and the updation agent. When the connection agent receives a request message, it makes the initial connection with the updation agent. For secure communication between the user agent and the connection agent, the proposed technique introduces security agent 1.

Security agent 1: Security agent 1 is employed for secure communication between the user and the connection agent by means of a cryptographic algorithm. Here, we are using the ECC method for encrypting the user symptoms.

The detailed process of the proposed security agent using cryptography algorithm is illustrated in the following section.

4.1 Elliptic Curve Cryptography

For security purpose, the recommended method utilizes ECC to encrypt the user symptoms. The main objective of the proposed technique is to prescribe the drugs for diabetes through secure communication.

In the proposed ECC algorithm, the private and public keys are produced by ECC, making the encrypted data safer. Here, user symptoms are encrypted for the ECC algorithm. The general equation of the elliptic curve is

$$y^2 = x^3 + ax + b. \tag{1}$$

To create both the public and private keys, we need key generation. To encrypt the message, the sender will be using the receiver's public key and the receiver will decrypt using the private key. The ECC algorithm proceeds through three fundamental phases:

1. Key generation: In this phase, the proposed public and private keys are selected. Here, the key generation is done using prime numbers.
2. Encryption: In this step, each user symptom is encrypted. Here, user symptoms are the input for the encryption method. The input user symptoms are encrypted and the output is split into two cipher text, C_1 and C_2.
3. Decryption: In the decryption process, the private key is used to decrypt the user symptoms.

The pseudo-code for ECC encryption algorithm is described below:

Pseudo-code for ECC encryption
Initialization: General equation of the elliptic curve: $y^2 = x^3 + ax + b$
1. Key Generation
Input: Select the random prime numbers.
Output: Public key (p_k) and private key (r_n).
Procedure:
– Select the random number (r_n)
– Generate the public key $p_k = r_n * P$
r_n – random values
P – point of the curve
p_k – public key
2. Encryption
Input: User symptoms.
Output: Cipher text C_1 and C_2.
Procedure:
– Collect the user symptoms.
– Compute cipher text by splitting two messages (C_1 and C_2):
$C_1 = r_n * P$
$C_2 = U_s + r_n * P$
r_n – random prime value
U_s – user symptoms
3. Decryption
Input: Cipher text C_1 and C_2
Output: User symptoms
Procedure:
– Get the cipher text C_1 and C_2
– Estimate the user symptoms: $U_S = C_2 - r_n * C_1$

Based on the above encryption, we encrypt the user symptoms. The decryption part is done by the connection agent. For the decryption, the connection agent needs the user public key. Based on the user public key, the connection agent decrypts the user symptoms and then makes the initial connection with the updation agent.

Updation agent: An updation agent is responsible for prescribing drugs to the corresponding user based on user symptoms. Here, we need also secure communication because the proposed technique introduces another security agent between the connection agent and the updation agent.

Security Agent 2: Security agent 2 is employed for secure communication between the connection agent and the updation agent through a cryptographic algorithm. Here, we are using OAES for the encryption. The connection agent key is used for encrypting the user symptoms, which will be optimally selected by means of the DE algorithm.

The detailed process of the proposed security agent using optimal cryptography algorithm is illustrated in the following section.

4.2 Optimal Advanced Encryption Standard

AES is a block cipher with a block length of 128 bits [15]. It allows three different key lengths: 128, 192, or 256 bits. We propose AES with 128-bit key length. The encryption process consists of 10 rounds of processing for 128-bit keys. Except for the last round in each case, all other rounds are identical. The 4×4 matrix of bytes made from the 128-bit input block is referred to as the state array. The different transformation operates on the intermediate results, known as a state; the state is basically in the form of a rectangular array of bytes. Before any round-based processing encryption can begin, input state is XOR with the first four words of the schedule.

A state of the proposed work is represented as

$$\begin{bmatrix} U0,0 & U0,1 & U0,2 & U0,3 \\ U1,0 & U1,1 & U1,2 & U1,3 \\ U2,0 & U2,1 & U2,2 & U2,3 \\ U3,0 & U3,1 & U3,2 & U3,3 \end{bmatrix}$$

A key value of the proposed work is represented as

$$\begin{bmatrix} K0,0 & K0,1 & K0,2 & K0,3 \\ K1,0 & K1,1 & K1,2 & K1,3 \\ K2,0 & K2,1 & K2,2 & K2,3 \\ K3,0 & K3,1 & K3,2 & K3,3 \end{bmatrix}$$

Here, we optimally select the key values based on the optimization technique and the DE algorithm is used to select the optimal key values. The step-by-step procedure of the DE algorithm is described in the following section.

4.2.1 Differential Evaluation Algorithm

The DE algorithm is a population-based algorithm like genetic algorithms using the similar operators: crossover, mutation, and selection [9]. The main difference in constructing better solutions is that genetic algorithms rely on crossover, whereas DE relies on mutation operation. The major task invariably depends on the divergences of arbitrarily sampled couples of solutions in the population. The novel technique employs the mutation function as a search mechanism and the selection function to manage the search for the potential zones in the search space. Further, it utilizes a non-uniform crossover that is capable of taking the child vector parameters from one parent more frequently than in the case of others. Using the components of the existing population members to construct trial vectors, the crossover operator efficiently shuffles the information about successful combinations, enabling the search for a better solution space.

The vital operators involved in the innovative method are the following:

1. Initialization: At the outset, let us suppose that the solution comprises n individuals. K_i represents the ith individual of the solution. Here, each solution represents the key value. At first, the initial solution is chosen arbitrarily.

$$K_i = \{k_1, k_2, \ldots k_n\}, \tag{2}$$

where n corresponds to the number of individuals.

2. Mutation: There is a number of methods for the mutation of individuals in differential evolution. As a rule, the mutation vector is generated through

$$M_{i+1} = K_i + CF(k_{r_1} - K_i) + SF(k_{r_2} - k_{r_3}), \tag{3}$$

where i and $r_1, r_2, r_3 \in \{1, 2, \ldots n\}$, CF is the combination factor, and SF is the scaling factor.

3. Crossover: The parent vector is blended with the mutated vector to generate a new vector.

$$N_{ji+1} = \begin{cases} M_{ji+1} & \text{if (rand} \le CR\text{) or } j{=}m \\ S_{ji} & \text{if (rand} > CR\text{) and } j{\neq}m \end{cases} \tag{4}$$

where $j = 1, 2, \ldots D$, rand $\in [0, 1]$, CR is the crossover rate $\in [0, 1]$, and $m \in (1, 2, \ldots D)$.

4. Selection: The entire solutions in the population have an identical option of being shortlisted as the parents regardless of their fitness values. The child generated after the mutation and crossover functions is subjected to assessment. Subsequently, the performance of the child vector and its parent are assessed and contrasted, choosing the superior one. In the event of the parent maintaining the superior position, it continues to sustain the population. Thus, the optimized value will be examined for the key value. Based on the optimal key value, the encryption and decryption processes are carried out.
   a. Encryption: For encryption, each round consists of the following four steps:
      1. Sub-byte operation: This is a non-linear byte substitution, independently operating on each byte of the state. We utilize the pre-calculation when the S-box is independent of any input. Then, we substitute each byte of the state in the s-box whose index corresponds to the value in the state.
      2. Shift row operation: In this operation, every row of the state is cyclically shifted to the left, which depends on the row index.
      3. Mix-column operation: This transformation operates on the state column by column, treating each column as a four-term polynomial. The purpose of this step is to provide the diffusion of the bits over multiple rounds. This is achieved by performing multiplication one column at a time. Each value in the column is multiplied by every row value of a standard matrix.
      4. Add round key: In Add Round Key, we apply round key to the state by bitwise XOR. The round key can be derived from the cipher key using key schedule.

      Based on the above encryption technique, we encrypt the user symptoms. Here, the decryption part is done in the updation agent. For decryption, the updation agent needs the connection agent key. Based on the connection agent key, the updation agent decrypts the user symptoms and then prescribes the drugs for the corresponding user based on the user symptoms.
   b. Decryption: In this mode, the operations are in reverse order compared with their order in the encryption mode. Thus, it starts with an initial round, followed by nine iterations of an inverse normal round and ends with an add round key. An inverse normal round consists of the following operations in this order: add round key, inverse mix columns, inverse shift rows, and inverse sub-bytes. From that process, we decrypt the user symptoms in an effective manner. After that, we have to classify the level of diabetes (normal or abnormal) of the user on the basis on that only the updation agent prescribes the drugs for the corresponding user. Our proposed technique uses the MKSVM algorithm for classifying the diabetes level. The detail process of the MKSVM is clearly illustrated in the following.

4.3 Classification Using Multiple Kernel Support Vector Machine

The user symptoms are furnished to enhance the support vector machine for the purpose of diabetes level classification. Here, the decrypted output from the earlier process is effectively employed for the segregation of the two classes. The SVM approach seeks the optimal separating hyper-plane between the classes by

focusing on the training cases that are placed at the edge of the class descriptors. These training cases are called the support vectors. The training cases other than the support vectors are the discarded vectors. SVM has confirmed its competence over the neural networks and the radial basis function (RBF) classifiers. Unlike neural networks, this model builds and does not require a hypothesized number of neurons in the middle layer or defining the center of the Gaussian functions in RBF. The SVM employs an optimum linear separating the hyper-plane to divide the two sets of data in a feature space. By maximizing the minimum margin between the two sets, the relative optimum hyper-plane is generated. As a result, the resulting hyper-plane will only rely on the border training patterns called the support vectors. To perform the non-linear process, the kernel functions are initiated in the SVM classification. The kernel methods are a class of algorithms for the pattern analysis whose best-known member is the SVM. A kernel is a similarity function, which serves as the domain expert, is provided to a machine learning algorithm. The kernel functions have been introduced for the sequential data, graphs, text, images as well as the vectors. There are two vital stages in the SVM process such as the training and testing stages.

Training phase: The decrypted output is furnished as the input of the training phase. The input function gives the set of values that cannot be separated. Almost all the potential segregations of the point set are realized by means of a hyperplane. In the Lagrange configuration, it is possible to locate the separation of the hyperplane normal vector through the dissimilar kernel function. In this connection, a kernel represents any function that relates to a dot product for a certain type of feature mapping. Nevertheless, mapping a point to a superior dimensional space is likely to lead to excessive evaluation duration and huge storage needs. Hence, in actual practice, a novel kernel function capable of directly evaluating the dot product in the superior dimensional space is introduced. The common version of the kernel function is

$$K(a, b) = \varphi(a)^T \varphi(b). \tag{5}$$

In this regard, the most extensively employed kernel functions include the linear kernel, polynomial kernel, quadratic kernel, sigmoid and the RBF. Given below are the expressions for the various kernel functions.

For the linear kernel,

$$\text{linear}_k(a, b) = a^T b + c, \tag{6}$$

where a, b represents the inner products in the linear kernel and c is a constant.

For the quadratic kernel,

$$\text{quad}_k(a, b) = 1 - \frac{||a-b||^2}{||a-b||^2 + c}, \tag{7}$$

where a, b are the vectors of the polynomial kernel function in the input space.

For the polynomial kernel,

$$\text{poly}_k(a, b) = (\lambda a^T b + c)^e, \lambda > 0. \tag{8}$$

For the sigmoid kernel:

$$\text{sig}_k(a, b) = \tanh(\lambda a^T b + c), \lambda > 0. \tag{9}$$

The efficiency of SVM invariably relies on the choice of the kernel. In the event of the feature space being linearly inseparable, it has to be mapped into a superior dimensional space by means of the RBF kernel so that the issue will emerge as linearly separable. Moreover, the combination of any two kernel functions can yield higher precision than that obtained using any single kernel function.

MKSVM: In the innovative technique, a novel MKSVM devoted to the significant enhancement in the classification procedure is envisaged. Here, two kernel functions, such as linear and quadratic kernel functions, are combined to yield superb performance ratios. By combining Equations (10) and (11), the average

is estimated as suggested in the novel method. The combined kernel function is effectively employed in the MKSVM, and the average of the kernel function $\text{avg}_k(a, b)$ is

$$\text{avg}_k(a, b) = \frac{1}{2}(\text{lin}_k(a, b) + \text{quad}_k(a, b)). \tag{10}$$

$$\text{avg}_k(a, b) = \frac{1}{2}\left((a^T b + c) + \left(1 - \frac{\| a - b \|^2}{\| a - b \|^2 + c}\right)\right). \tag{11}$$

In the enhanced support vector machine, two kernels such as the linear and quadratic are taken into account to identify the hyperplane. By combining the two results, the average of the results is achieved and utilized to identify the hyperplane.

Testing phase: The output from the decryption process is furnished as an input to the testing phase and the output indicates the diabetes level. Based on the diabetes level, the updation agent prescribes the drugs for the corresponding user. The performance of the proposed work is evaluated in Section 5.

5 Results and Discussion

This section gives the detailed view of the result that is obtained by our proposed method of multi-agent-based diabetes diagnosis, and classification is performed on the working platform of JAVA. To improve the classification accuracy, the proposed technique uses the MKSVM. To develop a secure communication through an agent, we introduce two types of security agent with cryptography algorithm.

5.1 Data Set Description

The proposed method experiments with the Cleveland data set. This data set is taken from the UCI Machine Learning Repository (available at https://archive.ics.uci.edu/ml/datasets/Heart+Disease). This database contains 76 characteristics; however, all distributed tests refer to the use of a subset of 14 characteristics. Especially, ML researchers use only the Cleveland database until today. The "goal'" field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence) to 4. Experiments with the Cleveland database have concentrated on simply attempting to distinguish presence (values 1, 2, 3, 4) from absence (value 0). The names and social security numbers of the patients were recently removed from the database and replaced with dummy values. Six of the examples have been discarded because they had missing values. Class distributions are 54% heart disease absent and 46% heart disease present.

5.2 Performance Analysis

The performance analysis of our proposed technique is shown in the following section. Here Table 1 shows the various iteration and corresponding encryption and decryption time for both ECC and OAES encryption algorithm. In our method, we take the number of iteration as 10, 20, 30 and 40.

To encrypt the user symptoms, our proposed method using ECC takes 8456 ms for encryption and 5462 ms for decryption. By varying the number of iterations like 20, 30 and 40, encryption and decryption times are also varied. Here, 8569, 8647 and 8697 ms are taken for encryption and 5462 ms to obtain for decryption by varying iteration 20, 30 and 40. Using OAES, the encryption time is 9456, 9563, 9756 and 9987 ms for various iteration values such as 10, 20, 30 and 40. Similarly, the decryption time is 4658 ms for various iteration values like 10, 20, 30 and 40. The graphical representation of the proposed encryption and decryption time using ECC and OAES by varying the iteration is shown in Figures 2 and 3.

Table 1: Encryption and Decryption Time for Various Iteration.

No. of iteration	ECC		OAES	
	Encryption time	Decryption time	Encryption time	Decryption time
10	8456	5462	9456	4658
20	8569	5462	9563	4658
30	8647	5462	9756	4658
40	8697	5462	9987	4658

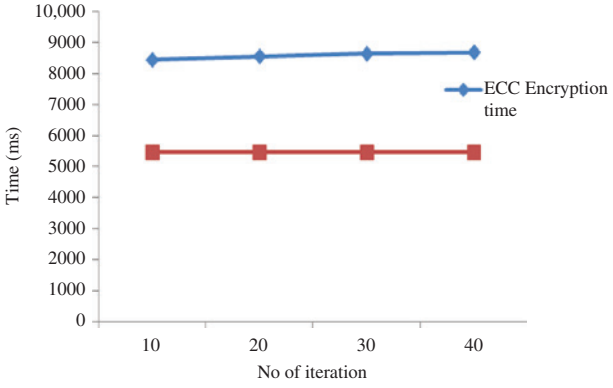

Figure 2: Encryption and Decryption Time by Varying the Number of Iteration Using ECC.

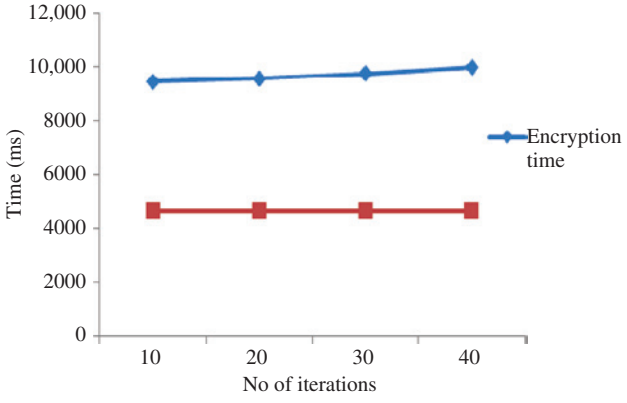

Figure 3: Encryption and Decryption Time by Varying the Number of Iteration Using OAES.

In our proposed technique, Table 2 shows the overall memory value and execution time of the proposed method. We vary the number of iteration and evaluate the memory value and execution time.

Figures 4 and 5 show the graph value for the number of iteration with memory value and execution time. For different iteration values such as 10, 20, 30 and 40, the execution time is 20,185, 21,654, 21,752, and 22,124 ms, whereas the memory value is 2,147,587, 2,236,547, 2,245,687 and 2,364,574. It is plotted in the following section.

The overall memory value of the proposed method achieves 2,248,598.75 bit by varying the number of iteration, and the memory value varyies for a number of iteration. The overall execution time of the proposed methods achieves 21,428.75 ms. Figure 5 shows the execution time for the proposed method by varying the number of iteration.

The overall classification accuracy of the proposed MKSVM algorithm is given in Table 3. Here, the proposed MKSVM achieves 77.7% of the accuracy value.

Table 2: Memory Value and Execution Time of the Proposed Method.

No. of iterations	Execution time	Memory value
10	20,185	2,147,587
20	21,654	2,236,547
30	21,752	2,245,687
40	22,124	2,364,574

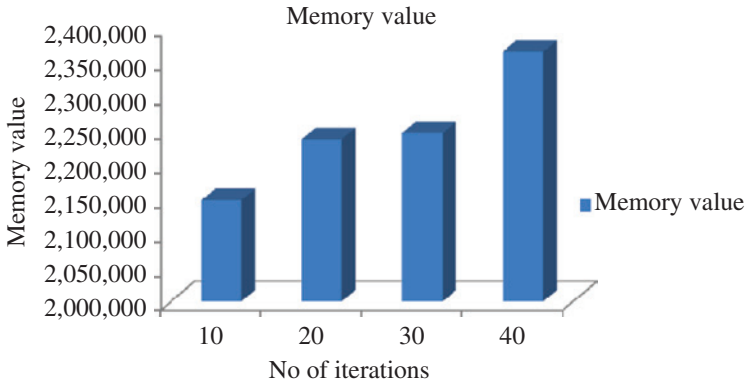

Figure 4: Memory Value for the Proposed Technique.

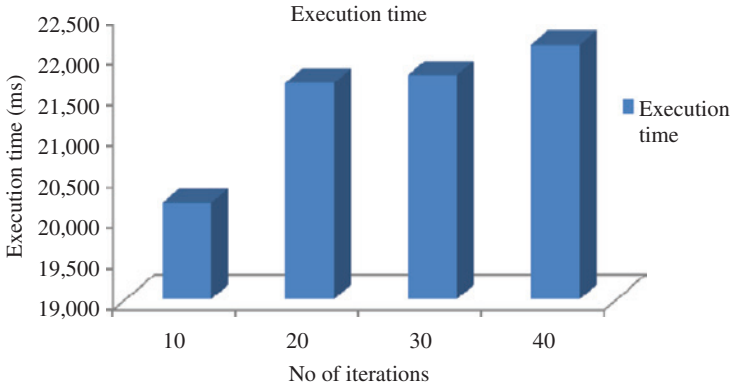

Figure 5: Execution Time for the Proposed Technique.

Table 3: Accuracy Value of Proposed Method by Varying the Iteration.

No. of iterations	Accuracy (%)
10	73.5
20	74.7
30	78.8
40	85.6

By varying the number of iteration, the proposed classification accuracy also changes. The proposed method achieves an accuracy of 73.5% for the 10th iteration, 74.7% for the 20th iteration, 78.8% for the 30th iteration and 85.6% for the 40th iteration. From the result, we analyze that if the proposed technique reaches the maximum iteration, it achieves better classification accuracy.

5.3 Comparative Analysis

The proposed method is compared with the existing method and the result is plotted given below. Table 4 shows the classification accuracy, sensitivity, specificity, precision and recall of the proposed method compared with existing method [14] and HFANN [21]. The graphical representation of the proposed comparative analysis of sensitivity, specificity, precision and recall values are represented in Figures 6–9.

Table 4 shows that the overall accuracy value of the proposed method is 78.15%, which is high when compared to the existing method. The sensitivity value of the proposed method is 91%, and the existing method achieves a sensitivity value of 90% and 88%. The specificity value for the existing method attains 50% and 53%, but the recommended technique attains maximum specificity value. The precision and recall value of the proposed methods are 86% and 91%, respectively. In this method, the proposed value is the maximum value when compared to the existing method. The graphical representation of the proposed comparative analysis of accuracy value is plotted in Figure 10.

5.3.1 Comparison Analysis Using Execution Time and Memory Value

The proposed execution time and memory value are compared with those of existing technique [21] (Figures 11 and 12).

When analyzing Figure 11, the proposed execution time is compared with the existing algorithm by varying the number of iteration. For iteration 10, the proposed technique takes 20,185 ms to complete, but the existing technique takes 2,156,841 ms, which is the maximum value when compared to the proposed method. The proposed technique takes 21,654 ms to complete the process for iteration 20. For iterations 30 and 40. The proposed technique needed shorter minimum time when compared to the existing methods. From the results, it is clear that the suggested technique takes minimum time for every iteration when compared to the existing methods.

Table 4: Comparative Analysis of Proposed Performance.

Methods	Existing method [14]	Existing method (HFANN) [21]	Proposed method (MKSVM)
Accuracy	0.75	0.77	0.78
Sensitivity	0.88	0.90	0.91
Specificity	0.50	0.53	0.55
Precision	0.77	0.84	0.86
Recall	0.87	0.89	0.91

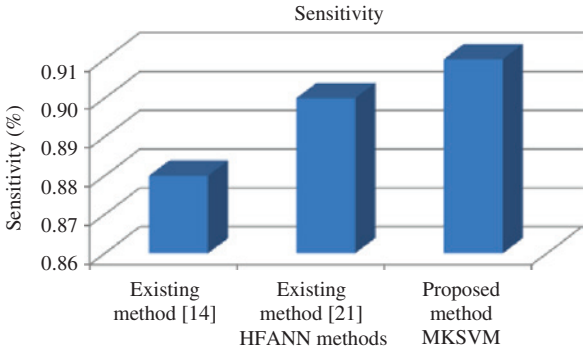

Figure 6: Comparative Analysis of Sensitivity.

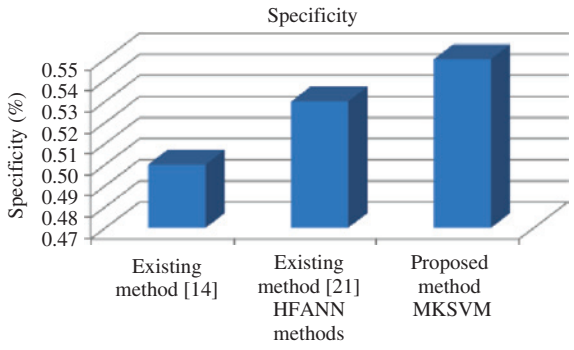

Figure 7: Comparative Analysis of Specificity.

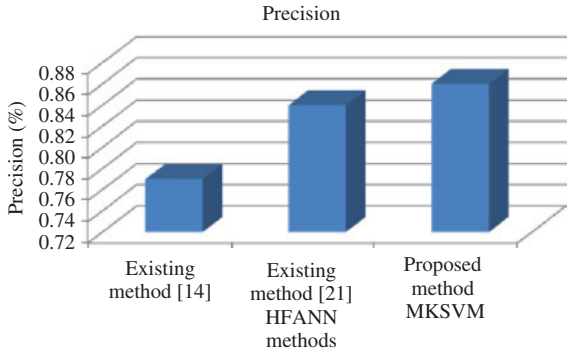

Figure 8: Comparative Analysis of Precision.

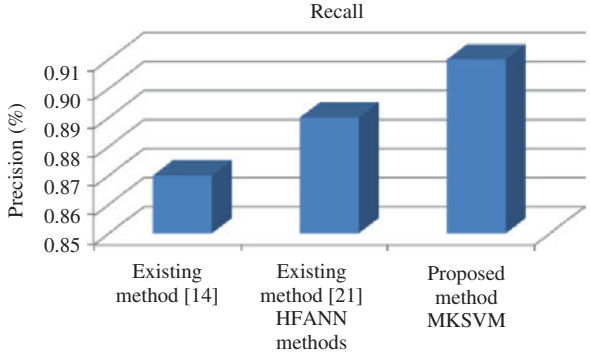

Figure 9: Comparative Analysis of Recall.

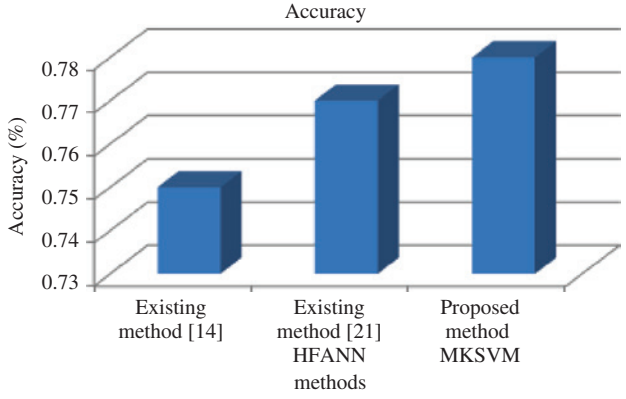

Figure 10: Comparative Analysis of Accuracy.

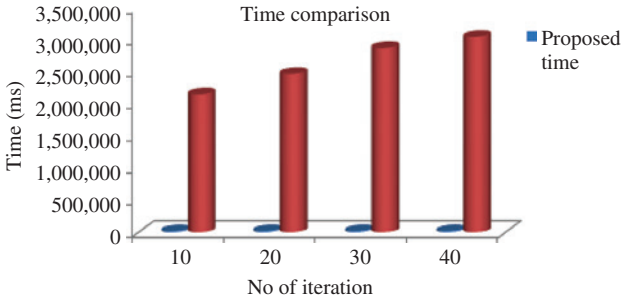

Figure 11: Comparative Analysis of Execution Time.

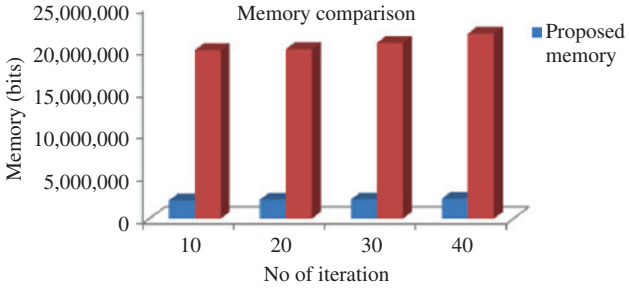

Figure 12: Comparative Analysis of Memory.

When analyzing Figure 12, the proposed memory value is compared with the existing algorithm by varying the number of iteration. For iteration 10, the proposed technique utilizes the memory value of 2,147,587 bits, but the existing technique takes 19,875,413 bits, which is the maximum value when compared to the proposed method. The proposed technique utilizes 2,236,547-bit memory value for iteration 20. For iterations 30 and 40, the proposed technique using the memory is the minimum value when compared to the existing methods. From the results, it is clear that the suggested technique utilizes minimum memory value for each iteration when compared to the existing methods.

5.3.2 Comparison Analysis Using Key Breaking Time

The key breaking time is an essential one to ensure the duration needed by the hackers to crack the key and get access to the secured data. As the duration of key breaking gets decreases, the protection of the data sets increases; our proposed method has higher key breaking time when compared to the existing methods. Hence, the proposed method assures high security. Table 5 shows the key breaking time comparison for the existing and proposed method.

From Table 5, it is clear that that the duration taken for key breaking in the proposed method is more than the duration taken for key breaking in the existing method. In our proposed technique, to break the key value in security agent 1, the hacker tries 135 times but the existing RSA technique the key would be broken in 115 times, which is the minimum number of times when compared to our implemented technique. In security

Table 5: Comparative Analysis of Key Breaking Time.

	Security agent 1		Security agent 2	
	Proposed (ECC)	**Existing (RSA)**	**Proposed (OAES)**	**Existing (AES)**
Maximum key breaking times	135	115	176	165

agent 2, the proposed key breaks at 176 tries, whereas using the existing method, the key breaks at 165 tries. Thus, the proposed method serves the best security. From all the above results, it is clear that the proposed method has high classification accuracy and high security than the existing methods.

6 Conclusions

Secure communication through the multi-agent system-based diabetes diagnosing and classification system is proposed in this paper. The proposed technique is implemented in JAVA platform. The performance of the proposed technique is evaluated using classification accuracy, sensitivity, specificity, precision, recall, execution time and memory value. The classification accuracy of the proposed technique is compared with the existing method. From the result, the classification accuracy of the implemented technique is high when compared to the existing classifier technique. The proposed multi-agent-based diabetes diagnosis achieves a maximum classification accuracy of 78.15%, sensitivity of 91%, specificity of 55%, and precision and recall of 86% and 91%, respectively. Our proposed method has higher key breaking time when compared to the existing methods. Thus, the proposed method serves the best security. In the future, the researcher will have sufficient opportunities to perform effective classifier to improve the classification accuracy and produce very excellent performance.

Bibliography

[1] R. P. Ambilwade, R. R. Manza and B. P. Gaikwad, Medical expert systems for diabetes diagnosis: a survey, *IJARCSSE* **4** (2014), 647–652.
[2] S. Chakraborty and S. Gupta, Medical application using multi agent system – a literature survey, *Int. J. Eng. Res. Appl.* **4** (2014), 528–546.
[3] W.-T. Chang, M.-C. Huang, H.-F. Chung, Y.-F. Chiu, P.-S. Chen, F.-P. Chen, C.-Y. Lee, S.-J. Shin, S.-J. Hwang, Y.-F. Huang and C.-C. Hsu, Interleukin-6 gene polymorphisms correlate with the progression of nephropathy in Chinese patients with type 2 diabetes: a prospective cohort study, *Diabetes Res. Clin. Pract.* **120** (2016), 15–23.
[4] J.-F. Chen, C.-M. Chang, M.-C. Kuo, S.-C. Tung, C.-F. Tsao and C.-J. Tsai, Impact of baseline body mass index status on glucose lowering and weight change during sitagliptin treatment for type 2 diabetics, *Diabetes Res. Clin. Pract.* **120** (2016), 8–14.
[5] M. Dalal, M. Grabner, N. Bonine, J. J. Stephenson, A. DiGenio and N. Bieszk, Are patients on basal insulin attaining glycemic targets? Characteristics and goal achievement of patients with type 2 diabetes mellitus treated with basal insulin and physician-perceived barriers to achieving glycemic targets, *Diabetes Res. Clin. Pract.* **121** (2016), 1–36.
[6] Z. Dong, Y. Wang, Q. Qiu, X. Zhang, L. Zhang, J. Wu, R. Wei, H. Zhu, G. Cai, X. Sun and X. Chen, Clinical predictors differentiating non-diabetic renal diseases from diabetic nephropathy in a large population of type 2 diabetes patients, *Diabetes Res. Clin. Pract.* **121** (2016), 1–26.
[7] S. Gupta, A. Sarkar, I. Pramanik and B. Mukherjee, Implementation scheme for online medical diagnosis system using multi agent system with JADE, *Int. J. Sci. Res.* **2** (2012), 1–6.
[8] B. Hashemi and H. Javidnia, An approach for recommendations in self management of diabetes based on expert system, *IJCA* **53** (2012), 6–12.
[9] M. Iwan, R. Akmeliawati, T. Faisal and H. M. A. A. Al-Assadi, Performance comparison of differential evolution and particle swarm optimization in constrained optimization, *Procedia Eng.* **41** (2012), 1323–1328.
[10] S. R. Kadu and A. D. Gawande, Review: multi-agent expert system for recommendations in self-management of diabetes, *IJARCSMS* **2** (2014), 259–263.
[11] C. Le, S. Rong, Y. Dingyun and C. Wenlong, Socioeconomic disparities in type 2 diabetes mellitus prevalence and self-management behaviors in rural southwest China, *Diabetes Res. Clin. Pract.* **121** (2016), 9–16.
[12] G. Maltoni, R. Minardi, C. P. Cristalli, L. Nardi, F. Alberton, V. Mantovani and S. Zucchini, A novel compound heterozygous mutation in an adolescent with insulin-dependent diabetes: the challenge of characterizing Wolfram syndrome, *Diabetes Res. Clin. Pract.* **121** (2016), 59–61.
[13] C. Neville, C. C. Patterson, G. J. Linden, K. Love, M. C. McKinley, F. Kee, S. Blankenberg, A. Evans, J. Yarnell and J. V. Woodside, The relationship between adipokines and the onset of type 2 diabetes in middle-aged men: the PRIME study, *Diabetes Res. Clin. Pract.* **120** (2016), 24–30.

[14] A. A. Obiniyi and M. K. Ahmed, Multi-agent based diagnostic model for diabetes, *IJSER* **6** (2015), 1589–1594.
[15] M. Pitchaiah, P. Daniel and Praveen, Implementation of advanced encryption standard algorithm, *IJSER* **3** (2012), 19–23.
[16] M. Pradhan and G. R. Bamnote, Predictive modeling of clinical data using soft computing – diabetes a case study, *Int. J. Comput. Commun.* **1** (2011), 31–37.
[17] F. Ranadive and P. Sharma, OpthoABM – an intelligent agent based model for diagnosis of ophthalmic diseases, *IJECS* **3** (2014), 9667–9670.
[18] S. Rani and D. Kumar, A case study on soft computing techniques used for diabetes mellitus, *IJARCSSE* **4** (2014), 1–4.
[19] H. Salem, G. Attiya and N. El-Fishawy, A survey of multi-agent based intelligent decision support system for medical classification problems, *IJCA* **123** (2015), 20–25.
[20] S. Su, C. Zhang, F. Zhang, H. Li, X. Yang and X. Tang, The association between leptin receptor gene polymorphisms and type 2 diabetes mellitus: a systematic review and meta-analysis, *Diabetes Res. Clin. Pract.* **121** (2016), 49–58.
[21] K. Tangod and G. Kulkarni, Multi agent based diabetes diagnosing and classification with the aid of hybrid firefly-neural network, *IJIES* (2016), 68–77.
[22] F. Tatari, M. R. Akbarzadeh T and M. Mazouchi, A self-organized multi agent decision making system based on fuzzy probabilities: the case of aphasia diagnosis, *J. Fuzzy Syst.* **11** (2014), 21–46.
[23] E. S. Viskum and M. L. Pedersen, Prevalence of diagnosed diabetes and quality of care among Greenlanders and non-Greenlanders in Greenland, *Diabetes Res. Clin. Pract.* **121** (2016), 1–18.
[24] W. Wang, M. He and W. Huang, Association of monocyte chemoattractant protein-1 gene 2518A/G polymorphism with diabetic retinopathy in type 2 diabetes mellitus: a meta-analysis, *Diabetes Res. Clin. Pract.* **120** (2016), 40–46.
[25] A. Willis, M. Roshan, N. Patel, L. J. Gray, T. Yates, M. Davies and K. Khunti, A community faith centre based screening and educational intervention to reduce the risk of type 2 diabetes: a feasibility study, *Diabetes Res. Clin. Pract.* **120** (2016), 73–80.