Optimal knowledge extraction technique based on hybridisation of improved artificial bee colony algorithm and cuckoo search algorithm

S. Jagadeesh Soundappan*

Department of CSE,
St. Peter's University, India
Email: jagadeeshsoundappan0777@gmail.com
*Corresponding author

R. Sugumar

Department of Computer Science and Engineering,
Velammal Institute of Technology,
Chennai, India
Email: sugu16@gmail.com

Abstract: We present a framework that we are currently developing, that allows one to extract knowledge from the knowledge discovery in database (KDD) dataset. Data mining is a very active and space growing research area. Knowledge discovery in databases (KDD) is very useful in scientific domains. In simple terms, association rule mining is one of the most well-known methods for such knowledge discovery. Initially, database are divided into training and testing for the aid of fuzzy generating the rules using fuzzy rules generation the set of rules are generated from the given dataset. From the generated rules, we are extracting the significant rules by using the improved artificial bee colony algorithm and cuckoo search algorithm (IABCCS). After extracting optimal knowledge from the dataset via rules, the data will be classified using fuzzy classifier with the aid of this finally we will classify the attack and normal.

Keywords: knowledge discovery; data mining; DM; scientific domains; fuzzy; association rule mining; artificial bee colony algorithm; cuckoo search; CS.

Reference to this paper should be made as follows: Jagadeesh Soundappan, S. and Sugumar, R. (xxxx) 'Optimal knowledge extraction technique based on hybridisation of improved artificial bee colony algorithm and cuckoo search algorithm', *Int. J. Business Intelligence and Data Mining*, Vol. X, No. Y, pp.000–000.

Biographical notes: S. Jagadeesh Soundappan obtained his Bachelor's degree in Computer Science from the Bannari Amman Institute of Technology, Sathyamangalam Tamil Nadu, India in 2001. And then, he obtained his Master's degree in Software Engineering from the Sri Ramakrishna Engineering College, Coimbatore Tamil Nadu in 2005. He is pursuing his PhD from St. Peter's University, from 2013 till date. He has also obtained his Oracle DBA Certifications. Currently, he is working as a Test Lead at Syntel International Pvt. Ltd., Chennai. His specialisations includes in data mining

R. Sugumar has received his BE degree from the University of Madras, Chennai, India in 2003, MTech degree from Dr. M.G.R. Educational and Research Institute, Chennai, India in 2007, and PhD degree from Bharath University, Chennai, India in 2011. From 2003 to 2015, he has worked at different levels in various reputed engineering colleges across India. He is currently working as an Associate Professor in the Department of Computer Science and Engineering at Velammal Institute of Technology, Chennai, India. His research interests include data mining, cloud computing and networks. He has published more than 30 research articles in various international journals and conference proceedings. He is acting as a reviewer in various national and international journals. He has chaired various international and national conferences. He is a Life Time Member of ISTE and CSI.

1 Introduction

Data mining (DM) is a very active and space growing research area in the field of computer science. Knowledge discovery in databases (KDD) is very useful in economic and scientific domains. Huge amount of data has been stored in documents or in the hard disks of computers. KDD techniques are used to reveal critical information hidden in the datasets (Fu and Wang, 2001). KDD has been a vigorous and attractive research challenge both in the areas of computing and DM. Its aim is to discover interesting and useful data from an oversized variety of data stored in the transactional databases. Rule extraction is a common task of KDD. Association rule mining is one of the most well-known methods for such knowledge discovery (Gupta and Sikka, 2013). It can effectively extract interesting relations among attributes from transactional databases to help out in decision-making. In some previous literature knowledge discovery is defined as the non-trivial process of discovering valid, novel, useful and interesting patterns in databases. This definition focuses on KDD as a complex process having variety of steps. DM is one such step during this process where intelligent techniques are applied so as to extract interesting data patterns. Most knowledge discovery or DM tools and techniques are based on statistics, machine learning, pattern recognition or artificial neural networks (Al-Maqaleh and Shahbazkia, 2012).

Genetic algorithm (GA) is an adaptive heuristic search technique purely based on the natural evolution process. It is widely used by the machine learning and DM community for the classification rule discovery process (Vivekanandan et al., 2013; Shivani et al., 2014). It imitates the mechanics of natural species evolution with biological science principles, like natural selection, crossover and mutation. It has been proved that the performance of the GA is comparable with the other classification methodologies like neural networks, decision trees, etc. GA searches for good solutions to a problem by maintaining a population of candidate solutions and making subsequent generations by choosing the current best solutions and using operators like crossover and mutation to create new candidate solutions (Mittal, 2012). Thus, better and better solutions are 'evolved' over time. Commonly, the algorithm terminates when either a maximum number of generations have been made, or a satisfactory fitness level has been reached for the population. The advantage of GA becomes clearer once the search space of a task is enormous (Gupta and Sikka, 2013).

Another evolution method used in DM is particle swarm optimiser (PSO). However, PSO has evolved in DM in which it can reduce complexity and speed up the DM process (Shukran et al., 2011). Although PSO shares many similarities with evolutionary computation techniques, the standard PSO does not use evolution operators such as crossover and mutation. PSO emulates the swarm behaviour of insects, animals herding, birds flocking, and fish schooling where these swarms search for food in a collaborative manner. Each member in the swarm adapts its search patterns by learning from its own experience and other members' experiences (Kadirkamanathan and Fleming, 2005). These phenomena are studied and mathematical models are constructed. In PSO, a member in the swarm, called a particle, represents a potential solution which is a point in the search space. The global optimum is regarded as the location of food. Each particle has a fitness value and a velocity to adjust its flying direction according to the best experiences of the swarm to search for the global optimum in the D-dimensional solution space (Liang et al., 2006; Ratnaweera et al., 2004).

Swarm intelligence is widely used to address selection of the optimal feature set. An artificial bee colony (ABC) algorithm is a recently introduced optimisation algorithms, to simulate intelligent foraging behaviour of honey bee swarm (Shanthi and Bhaskaran, 2014). Clustering analysis is used in many disciplines/applications to identify homogeneous groups of objects based on their attribute values. ABC was used for data clustering on benchmark problems and its performance was compared to a particle swarm optimisation (PSO) algorithm and nine other classification techniques from the literature. Thirteen typical test datasets from UCI Machine Learning Repository demonstrated techniques results. The simulation indicated that ABC algorithm was efficient for multivariate data clustering. The simulation results revealed that ABC algorithm's performance was comparable to the above mentioned algorithms and can be used to solve high dimensionality engineering problems (Sumathi et al., 2014; Kumar and Singh, 2013). Since the ABC algorithm is simple in concept, easy to implement, and has fewer control parameters, it has been widely used in many optimisation applications such as protein tertiary structures, digital IIR filters, artificial neural networks and others (Shukran et al., 2011; Subanya and Rajalaxmi, 2014).

2 Literature review

Papageorgiou (2010) has proposed a novel approach for the construction of augmented fuzzy cognitive maps based on DM and knowledge-extraction methods for decision-making and classification tasks. In this proposed method, the fuzzy cognitive map was a knowledge-based technique that works as an artificial cognitive network inheriting the main aspects of cognitive maps and artificial neural networks. Decision trees, in the other hand, are well known intelligent techniques that extract rules from both symbolic and numeric data. In this proposed method, they approached to incorporate any type of knowledge extraction algorithm. Furthermore, through the knowledge extraction methods the useful knowledge from data could be extracted in the form of fuzzy rules and inserted those into the FCM, contributing to the development of a dynamic approach for decision support.

Ceravolo et al. (2004) have proposed a fuzzy technique to compare XML documents belonging to a semi-structured flow and sharing a common vocabulary of tags. The

research used an approach based on the idea of representing documents as fuzzy bags and, using a measure of comparison, evaluating structural similarities between them. The paper had also suggested how to organise the extracted knowledge in a class hierarchy, choosing a technique related to the domain of interest, later to be converted into user ontology.

Nováček (2008) has aimed at bottom-up generation and merging of ontologies, which are related to ontology learning (OLE). The research applied expressive uncertain knowledge representation framework called adaptive net of universally interrelated concepts (ANUIC). They had discussed their recent research result in taxonomy acquisition and showed how even a simple application of the principles of ANUIC can improve the results of initial knowledge extraction methods. The paper had also suggested an algorithm for large-scale automatic annotation of natural language documents, applying uncertain knowledge bases created using our approach.

Pechenizkiy et al. (2003) have proposed a technique of three different eigenvector-based feature extraction approaches for classification. The summary of obtaining results concerning the accuracy of classification schemes was presented and the issue of search for the most appropriate feature extraction method for a given dataset was considered. A decision support system to aid in the integration of the feature extraction and classification processes was also proposed. The means of knowledge acquisition needed to build up the proposed system were considered.

Du et al. (2012) have proposed a rule extraction approach based on combing hybrid genetic double multi-group cooperative PSO algorithm and discrete PSO algorithm (named HGDMCPSO/DPSO) to discover the previously unknown and potentially complicated nonlinear relationship between key parameters and variances handling measures of CP. Then these extracted rules can provide abnormal variances handling warning for medical professionals. Three numerical experiments on Iris of UCI datasets, Wisconsin breast cancer datasets and CP variances datasets of osteosarcoma preoperative chemotherapy are used to validate the proposed method. Results showed that the proposed rule extraction algorithm can obtain higher prediction accuracy, less computing time, more stability and easily comprehended by users.

Rotshtein and Mityushkin (2001) have proposed a method of GAs. The proposed method was based on the operations of crossover, mutation, and selection of initial variants of solutions or so-called chromosomes, from which the most optimal solutions were subsequently chosen. The method was illustrated by a computer experiment consisting of the determination of knowledge of a nonlinear object with two input variables and one output variable.

Chia et al. (2006) have proposed a novel technique for neulonet learning by composing net rules using genetic programming. This proposed method was based on sequential covering approach for generating a list of neulonets, the straightforward extraction of human-like logic rules from each neulonet provides an alternate perspective to the greater extent of knowledge that can potentially be expressed and discovered, while the entire list of neulonets together constitute an effective classifier. The paper showed how the sequential covering approach was analogous to association-based classification, leading to the development of an association-based neulonet classifier. The empirical study showed that associative classification integrated with the genetic construction of neulonets performs better than general association-based classifiers in terms of higher accuracies and smaller rule sets. This was due to the richness in logic expression inherent in the neulonet learning paradigm.

Chow et al. (1999) have proposed a novel method for enforcing heuristic constraints on membership functions for rule extraction from a fuzzy/neural architecture. The proposed method not only ensured that the final membership functions conform to a priori heuristic knowledge, but also reduced the domain of search of the training and improves convergence speed. The proposed method was also described on a specific fuzzy/neural architecture; it was applicable to other realisations, including adaptive or static fuzzy inference systems.

3 Problem definition

1 By applying the existing techniques, it is challenging to mine for complete and enlightening knowledge in such complex data suited to real-life decision needs.

2 The traditional techniques generally find out homogeneous features from a single source of information while it is not efficient to mine for patterns uniting components from multiple data sources.

3 Genetic optimisation algorithm is limited because of random solutions and convergence, in other words this means that the entire population is improving, but this could not be said for an individual within this population.

4 To generate patterns, association rule mining is a major technique. On the other hand, as huge numbers of association rules which may be sometimes insignificant.

5 Mined preconditions (combinations of risk assessment values) for software projects were to fall into turmoil by means of a dataset consisting of a large number of risk evaluation variables.

6 Association rule mining is frequently very expensive and sometimes unfeasible to link multiple data sources into a single dataset.

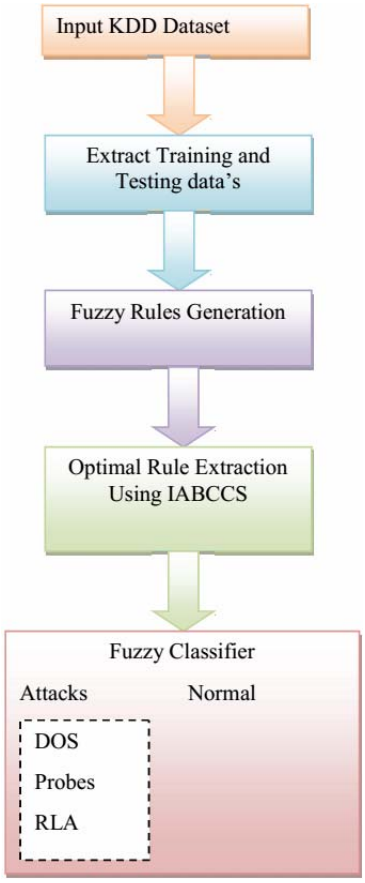
7 For extracting knowledge from different datasets using a single mining technique will not be efficient.

4 Proposed methodology

An active field of research that resolves the non-trivial process of associating valid, potentially useful, and ultimately understandable patterns in data is KDD. In simple terms, a pattern is actionable if the user can act upon it to the usage. Actionable patterns cannot only afford relevant grounds to business decision-makers for performing appropriate actions, but also deliver, expected outcomes to business. The main method to produce patterns is association rule mining, where a large numbers of association rules and insignificant rules are also often produced by association mining algorithm, which can be very tough for decision makers to not only understand such rules, but also identify a useful source of knowledge to apply to the business processes, but association rules can only give out limited knowledge for potential actions. For more informative and comprehensive knowledge for decision-making in the real world, there is a powerful and challenging need to mine. For discovering informative knowledge in complex data, a

comprehensive and general approach is suggested with the aid of fuzzy rule generation we will generate the rules, the set of rules are generated from the given dataset. From the set of rules, the significant rules are extracted by hybridisation of improved artificial bee colony algorithm (IABC) and cuckoo search (CS) algorithm. After extracting optimal knowledge from the dataset via rules, the data will be classified using fuzzy classifier.

Figure 1 Proposed IABCCS block diagram (see online version for colours)



4.1 KDD dataset

KDD is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data. KDD is an automatic, exploratory analysis and modelling of large data repositories. KDD is the organised process of identifying valid, novel, useful, and understandable patterns from large and complex datasets. DM is the core of the KDD process, involving the inferring of algorithms that explore the data, develop the model and discover previously unknown patterns. The model is used for understanding phenomena from the data, analysis and prediction. The accessibility and abundance of data today makes knowledge discovery and DM a matter of considerable

importance and necessity. Given the recent growth of the field, it is not surprising that a wide variety of methods are now available to the researchers and practitioners. No method is superior to others for all cases. The handbook of DM and knowledge discovery from data aims to organise all significant methods developed in the field into a coherent and unified catalogue; presents performance evaluation approaches and techniques; and explains with cases and software tools the use of the different methods. In our work, the KDD dataset contain 2,000 for training and 1,000 for testing. In that dataset, we are extracted three attacks and one normal, the attacks are DOS, Probe, and RLA.

4.1.1 Denial of service attacks

Denial of service (DOS) attack is an attack where the attacker creates a few calculations or memory resource completely engaged or out of stock to handle authentic requirements or reject justifiable users the right to utilise a machine. In this category, the attacker makes some computing or memory resources too busy or too full to handle legitimate request or deny legitimate users access to machine. DOS contains the attacks: 'Neptune', 'back', 'Smurf', 'pod', 'land', and 'teardrop'.

4.1.2 Probing attack (PROBE)

Probing (PROBE) is a collection of attacks where an attacker scrutinises a network to gather information or to conclude prominent vulnerabilities. In this category the attacker attempt to gather information about network of computers for the apparent purpose of circumventing its security. Probe contains the attacks: 'port sweep', 'Satan', 'Nmap', and 'Ip sweep'.

4.1.3 Random link attack

In an random link attack (RLA), the malicious user creates a set of false identities and uses them to communicate with a large, random set of innocent users. Attackers create some fake nodes and randomly connect to regular nodes. Fake nodes form some inner structure among themselves to evade detection.

From the dataset, we have taken 2,000 for training and 1,000 for testing then that data's are given in to the fuzzy rules generation to generate the rules, they are given below.

4.2 Fuzzy logic

Fuzzy rule-based classification is a method of generating a mapping from a given input to an output using fuzzy logic. Then, the mapping gives a basis, from which decisions can be generated. Membership functions, logical operations, and If-Then rules are used in the fuzzy rule-based process. The stages of fuzzy are,

1 fuzzification

2 fuzzy rules generation

3 defuzzification.

4.2.1 Fuzzification

During the fuzzification process, to convert the crisp input into linguistic variables. For the fuzzification process, the input is the best peak features, alpha mean and beat mean. After that, the minimum and maximum values are calculated from the input features and energy signals. The process of fuzzification is computed by applying the following format.

In our study, we are calculating the limit values we introduce

1 very low

2 low

3 medium

4 high.

The fuzzification module transforms the crisp input(s) into fuzzy values. These values are then processed in fuzzy domain by inference engine based on the knowledge base supplied by the domain expert(s). The knowledge base is composed of the rule base (RB), characterises the control goals and control policy of the domain expert by a set of linguistic control rules, and of the database (DB), containing the term sets and the membership functions defining their semantics. Finally, the processed output is transformed from fuzzy domain to crisp domain by defuzzification module.

4.2.2 Fuzzy rules generation

According to the fuzzy values for each feature that are generated in the fuzzification process, the fuzzy rules are also generated. The rules are

- *General form of fuzzy rule*

  'IF A THEN B'

  The 'IF' part of the fuzzy rule is called as 'antecedent' and also the 'THEN' part of the rule is called as 'conclusion'. The output values between 'Attack' and 'Normal' of the fuzzy is trained for generating the fuzzy rules.

- *Conditions:*

  1 if A value is 'Low' and B value is 'Low' then 'Very low'

  2 if A value is 'Low' and B value is 'Medium' then 'Low'

  3 if A value is 'Medium' and B value is 'High' then 'Medium'

  4 if A value is 'High' and B value is 'High' then 'High'.

If the fuzzy rules has been generated then using that rules we are generating the set of rules, the significant rules are extracted by hybridisation of IABC and CS algorithm.

4.3 Improved artificial bee colony – CS algorithm

4.3.1 Artificial bee colony

ABC algorithm is an optimisation algorithm based on the intelligent behaviour of honey bee foraging. This is a population-based optimisation technique. It is based on inspecting the behaviours of real bees on finding nectar amounts and sharing the information of food sources to the other bees in the hive. These specialised bees try to maximise the nectar amount stored in the hive by performing efficient. The honey bees from the colonies are extended over a very long distance in order to exploit the food sources in a multiple directions. The nectar sources with large amount of nectar that can be extracted with minimum difficulty are visited by more number of the bees while the nectar sources with less amount of nectar are discarded. The foraging begins by the scout bees which are sent to collect nectar. The search starts randomly by the scout bees from one source to another. When the scout bees reach the hive, they deposit the amount collected and goes to the dance floor to perform waggle dance. This type of dance is the sole medium of communicating the information about the most promising nectar source to all other bees in the hive. After watching the waggle dance, unemployed bees (onlooker bees) follow the scout bee to promising nectar source to collect nectar. This waggle dance process helps all the bees to collect nectar in efficient and faster way. The quality of food source is continuously monitored by the bees so as to propagate this information in the next waggle dance.

The three agents in ABC are:

- the employed bee
- the onlooker bee
- the scout bee.

The employed bees are associated with the specific food sources, the onlooker bees watching the dance of employed bees within the hive to choose a food source, and the scout bees searching for food sources randomly. The onlooker bees and the scout bees are the unemployed bees. Initially, the scout bees will discover the positions of all food sources, thereafter, the job of the employed bee starts. Artificial employed bees would probabilistically obtain some modifications on the position in its memory to target a new food source and find the nectar amount or the fitness value of the new source. Later, the onlooker bee evaluates the information taken from all artificial employed bees and then chooses a final food sources with the highest probability related to its nectar number. If the fitness value of the new one is higher than the previous one, then the bee forgets the old one and memorises the new position. This is called as greedy selection process. Then the employed bee whose food source has been exhausted becomes a scout bee to search for the further food sources once again.

4.3.2 Cuckoo search

CS algorithm is a metaheuristic algorithm which was inspired by the breeding behaviour of the cuckoos and alleviates to implement. There are a number of nests in CS. Each egg points out a solution and an egg of cuckoo indicates a new solution. The new and better solution is replacing the most awful solution in the nest. The subsequent representation

scheme is selected by CS algorithm: Each egg in a nest symbolises a solution, and a Cuckoo egg symbolises a novel solution. The plan is to employ the novel and probably better egg to substitute a not-so-good egg of Cuckoo in the nests. On the other hand this is the fundamental case, i.e., one cuckoo per nest, but the extent of the approach can be raised by incorporating the property that each nest can have more than one egg which symbolises a set of solutions.

- At a time only one egg is laid by cuckoo. Cuckoo dumps its egg in a arbitrarily selected nest.
- The number of accessible host nests is fixed, and nests with high quality of eggs will transmit over to the next generations.
- In case of a host bird found out the cuckoo egg, it can throw the egg away or discard the nest, and build a totally novel nest.

4.3.3 IABC algorithm steps

In IABC, the solutions represent the food sources and the nectar quantity of the food sources corresponds to the fitness of the associated problem. The number of employed and onlooker bees are same and this number is equal to the number of food sources. In our IABC, instead of onlooker bee we use cross over and mutation operation for the updation of solution randomly. The various steps involved in implementing IABC algorithm is explained below.

Step 1 Initialise the food source

The algorithm starts with randomly producing food source sites that correspond to the solutions in the search space. Produce the initial food source FS_i ($i = 1, 2, 3 \ldots n$) where n indicates the number of food source. This procedure is called initialisation process.

Step 2 Fitness evaluation

Using fitness function, the fitness value of the food source is computed to find the best food source. It is demonstrated as below,

$$fitness = \max PSNR \tag{1}$$

Step 3 Employed bee phase

In the employed bees phase, each employed bee finds a new food source FS_{ij}^{new} in the neighbourhood of its current source FS_i. The new food source is calculated using equation number (2).

$$FS_{ij}^{new} = FS_{ij} + \gamma\left(FS_{ij} - FS_{kj}\right) \tag{2}$$

where FS_{ij} is the j^{th} parameter of the i^{th} employed bee; FS_{ij}^{new} is a new solution for FS_{ij} in the j^{th} dimension; FS_{kj} is the neighbour bee of FS_{ij} in employed bee population; γ is a number arbitrarily chosen in the range of [–1, 1].

Step 4 Fitness evaluation for new food source

Fitness values are found for every new food source and choose the best food source.

Step 5 Greedy selection process

After choosing the best food source next use greedy selection process. Using the equation (3), find the probability of the chosen food source is calculated.

$$P_i = \frac{fitness_i}{\sum_{n=1}^{SN} fitness_n} \tag{3}$$

where $fitness_i$ is a fitness value of i^{th} employed bee.

Step 6 Instead of onlooker bee we use CS

After calculating the probability of the employed bee we will update values in CS algorithm. The steps are given below.

Step 1 *Initialisation phase:* The population (m_i, where $i = 1, 2, \ldots n$) of host nest is started arbitrarily.

Step 2 *Generating new cuckoo phase:* Using levy flights a cuckoo is selected at random and it produces new solutions. After that the produced cuckoo is evaluated using the objective function for finding out the quality of the solutions.

Step 3 *Fitness evaluation phase:* Evaluate the fitness function based on the equation and next select the best one.

$$fitness = maximum\ popularity \tag{4}$$

Step 4 *Updation phase:* Improve the initial solution by levy flights in which cosine transform is used. The superiority of the new solution is evaluated and a nest is selected among arbitrarily. If the excellence of new solution in the selected nest is better than the old solutions, it will be replaced by the new solution (Cuckoo). Otherwise, the previous solution is placed aside as the best solution. The levy flights employed for ordinary CS algorithm is,

$$m_i^* = m_i^{(t+1)} = m_i^{(t)} + \alpha \oplus Levy(n) \tag{5}$$

Step 5 *Reject worst nest phase:* The worst nests are thrown away in this part, based on their chance values and new ones are built. Presently, function the best solutions are ranked based on their fitness. After that the best solutions are identified and spotted as optimal solutions.

Step 6 *Stopping criterion phase:* Till the maximum iteration achieves this process is repeated. The optimised effect will be inspected for extract a significant rules.

4.3.4 Scout bee phase

The abandonment counters of all employed bees are tested with a number which is decided by designer (limit). The employed bee, which cannot improve self-solution until the abandonment counter reaches to the limit, becomes scout bee. The scout bee in which a solution was produced by itself to become a employed bee. Therefore, scout bees in ABC algorithm prevent stagnation of employed bee population.

Finally, utilising these rules we are extracted a significant set of rule from that rules we are extracting optimal knowledge from the dataset via. Rules, the data will be classified using fuzzy classifier

4.4 Fuzzy-based classification

Fuzzy classification is the process of grouping elements into a fuzzy set whose membership function is defined by the truth value of a fuzzy propositional function. In a fuzzy-based classification method used to classify the attack and normal, the fuzzy system will classify the three attacks they are DOS, Probe, RLA, and the remaining are normal. If the fuzzy system will generate a optimal rules then with the aid of IABCCS algorithm we will extract a significant rules and those rules are classified by using fuzzy classifier to classify a attack and normal. It is typically needed in fuzzy control systems. Some process of defuzzification is required to convert the resulting fuzzy set description of an action into a specific value for a control variable.

4.4.1 Defuzzification

Defuzzification is the process of producing a quantifiable result in fuzzy logic, given fuzzy sets and corresponding membership degrees. It is typically needed in fuzzy control systems these will have a number of rules that transform in to normal and attack. Fuzzy result, that is, the result is described in terms of membership in fuzzy sets. It is typically needed in fuzzy control systems. Defuzzification is the process of producing a quantifiable result in fuzzy logic, given fuzzy sets and corresponding membership degrees. It is typically needed in fuzzy control systems. Some process of defuzzification is required to convert the resulting fuzzy set description of an action into a specific value for a control variable. The input given for the Defuzzification process is the fuzzy set and the output obtained. As much as fuzziness supports the Rule Evaluation during the intermediate steps and the final output for every variable is usually a single number. The single number output is a value attack and normal. The attacks are DOS, PROBE, and RLA.

Finally, utilising this rules we are extracted a significant set of rule from that rules we are extracting optimal knowledge from the dataset via. Rules, the data will be classified using fuzzy classifier

5 Results and discussion

The experimental result of fuzzy-based classifier is discussed below. The proposed system is implemented using MATLAB 2014 and the experimentation is performed with i5 processor of 3GB RAM.

5.1 Dataset description

The proposed fuzzy-based classifier is experimented with the dataset namely KDD dataset. These dataset are given as input to identify the lunge nodules.

5.1.1 KDD dataset

This is the dataset used for The Third International Knowledge Discovery and DM Tools Competition, which was held in conjunction with KDD-99 (The Fifth International Conference on Knowledge Discovery and DM). The competition task was to build a network intrusion detector, a predictive model capable of distinguishing between 'bad' connections, called intrusions or attacks, and 'good' normal connections. This DB contains a standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment.

5.2 Evaluation metrics

An evaluation metric is used to evaluate the effectiveness of the proposed system. It consists of a set of measures that follow a common underlying evaluation methodology some of the metrics that we have choose for our evaluation purpose are True positive, True negative, False positive and False negative, Specificity, Sensitivity, Accuracy, F-measure.

5.2.1 Sensitivity

The measure of the sensitivity is the proportion of actual positives which are accurately recognised. It relates to the capacity of test to recognise positive results.

$$Sensitivity = \frac{TP}{(TP + FN)} \tag{6}$$

where *TP* stands for true positive and *FN* stands for false negative.

5.2.2 Specificity

The measure of the specificity is the extent of negatives which are properly recognised. It relates to the capacity of test to recognise negative results.

$$Specificity = \frac{TN}{(TN + FP)} \tag{7}$$

where *TN* stands for true negative and *FP* stands for false positive.

5.2.3 Accuracy

Accuracy of the proposed method is the ratio of the total number of TP and TN to the total number of data.

$$Accuracy = \frac{TN + TP}{(TN + TP + FN + FP)} \tag{8}$$

Table 1 condition and terms of TP, TN, FT, and FN

Experimental outcome	*Condition as determined by the standard of truth*	
	Positive	*Negative*
Positive	TP	FP
Negative	FN	TN

Table 2 Proposed and existing values of TP, TN, FT, and FN

Proposed			*Existing*		
TP	*TN*	*FP*	*TP*	*TN*	*FP*
4,227	574	554	2,765	1,488	1,636
4,518	914	488	4,328	535	17
5,155	466	314	4,340	985	83

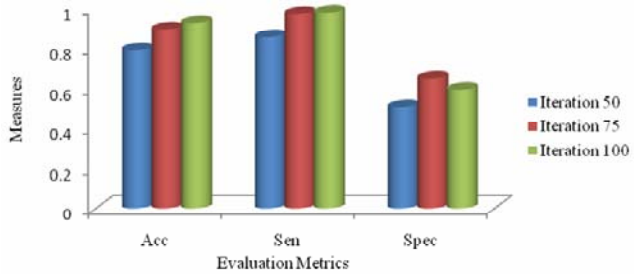
5.3 Performance analysis

The performance of the proposed heart disease prediction methods of human beings are evaluated by the three metrics sensitivity, specificity and accuracy. The results of proposed work help to analyse the efficiency of the prediction process. The subsequent Table 3 tabulates the results. Here, only the results of dataset given in Table 3.

Table 3 Results of the proposed lunge cancer disease prediction system with dataset

Sensitivity	*Specificity*	*Accuracy*
0.858797237	0.508865248	0.793553719
0.972030981	0.65192582	0.89785124
0.978178368	0.597435897	0.929090909

Figure 2 Graph for results with the performance measures specificity and sensitivity, accuracy



From Table 3, the evaluation metrics are analysed for the dataset, by which we can observe the efficiency of proposed detection system. The results of the measures sensitivity, specficity, and accuracy are graphically represented in Figure 2. The sensitivity of three iterations is 0.858797237%, 0.972030981%, 0.978178368%. With

these metrics, the specificity and accuracy are the main measures for evaluating the detection accuracy of our proposed system. The values of specificity for every iteration are 0.508865248%, 0.651925825%, 0.597435897% and the values of accuracy is 0.793553719%, 0.89785124%, 0.929090909%. The results get high accuracy results on behalf of the reduced error rates in the proposed system. From Figure 2 also, we find out the minimal value of error rates for the four dataset.

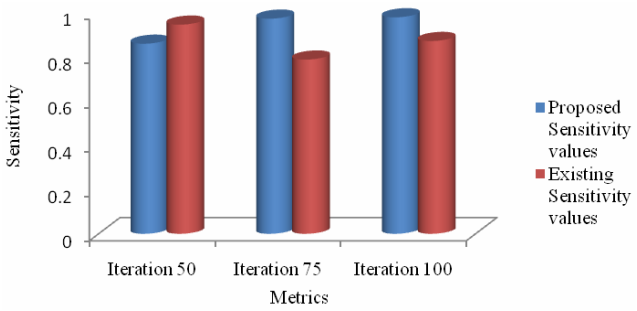
5.4 Comparative analysis

The literature review works are compared in this section with the proposed work to show that our proposed work is better than the state-of-art works. We can establish that our proposed work helps to attain very good accuracy for the attack prediction of DB using fuzzy. And also, we can establish this prediction accuracy outcome by comparing other classifiers. We have utilised ABC for our comparison in our work. The comparison outcomes are presented in Table 4.

Table 4 Comparison of sensitivity values in proposed vs. existing method

Iteration	*Proposed sensitivity values*	*Existing sensitivity values*
50	0.858797237	0.944976077
75	0.972030981	0.787195344
100	0.978178368	0.87113609

Figure 3 Comparison of proposed method vs. existing method for sensitivity (see online version for colours)

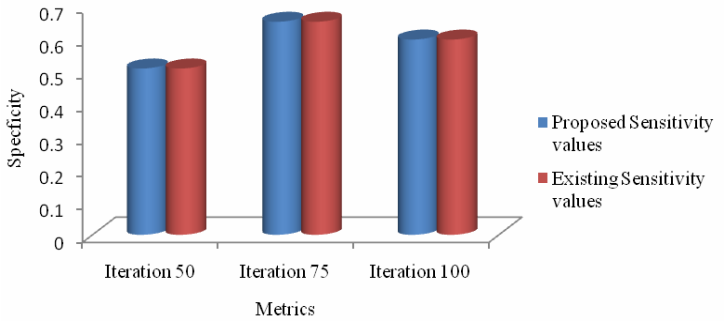


The sensitivity for the ABC is 0.944976077, 0.787195344, and 0.87113609 which is low in compared with our classifier, fuzzy for our dataset are 0.858797237, 0.972030981, and 0.978178368.

Table 5 Comparison of specificity values in proposed vs. existing method

Iteration	*Proposed specificity values*	*Existing specificity values*
50	0.508865248	0.508865248
75	0.65192582	0.65192582
100	0.597435897	0.597435897

Figure 4 Explains the comparison outcomes of the specificity measures with existing metrics (see online version for colours)

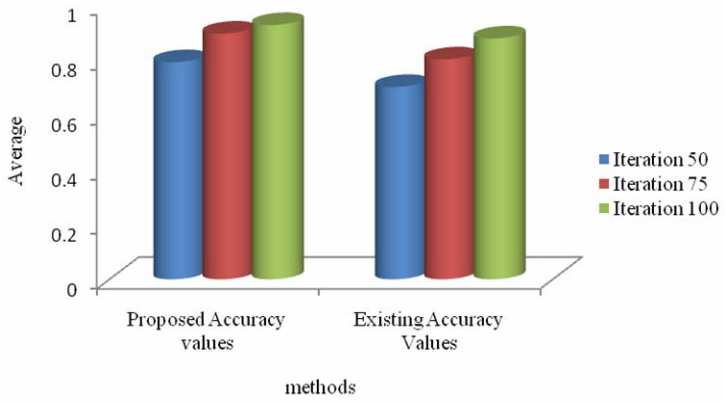


The specificity for the ABC are 0508865248, 0.65192582 and 0.597435897, which is equal in compared with our classifier, fuzzy for four dataset are 0508865248, 0.65192582 and 0.597435897.

Table 6 Comparison of accuracy values in proposed vs. existing

Iteration	*Proposed accuracy values*	*Existing accuracy values*
50	0.793553719	0.702975207
75	0.89785124	0.803801653
100	0.929090909	0.880165289

Figure 5 Explains the comparison outcomes of the accuracy with existing metrics and proposed metrics (see online version for colours)

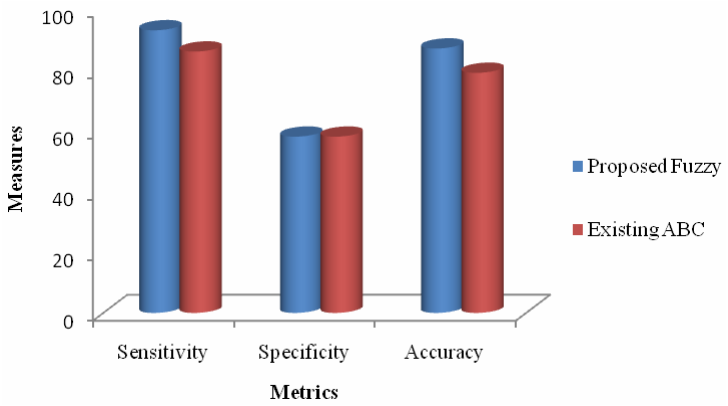


The accuracy for the ABC are 0.702975207, 0.803801653, and 0.880165289, which is low in compared with our classifier, fuzzy for four dataset are 0.793553719, 0.89785124, and 0.929090909.

Table 7 Comparison of proposed method vs. existing method

Metrics	*Proposed fuzzy in %*	*Existing ABC%*
Sensitivity	93	86
Specificity	58	58
Accuracy	87	79

Figure 6 Explains the comparison outcomes of the proposed method with existing method



The improved good accuracy outcomes of attack classification are presented by our proposed work. In comparison with the ABC gives very less accuracy values for the evaluation measures. The sensitivity values of ABC gives 86% but our proposed fuzzy gives 93%. The specificity values of existing ABC gives 58% which is equal to our proposed method. The accuracy values of existing ABC gives 79% but our proposed fuzzy gives 87%. From these outcomes, it is known that by means of fuzzy classifier in our work provides very good solution in classifying the attack and normal as it gives improved accuracy outcomes.

6 Conclusions

A fuzzy-based classification with three phases – fuzzy rule generation, optimal rule extraction and classification was proposed in this paper. Rules are generated based on the dataset then these rules are generated by using fuzzy rule generation and from that rules they are extracting some optimal rules by using IABCCS and finally rules are given into the fuzzy system. The performance measures of sensitivity, specificity, accuracy, were evaluated for our proposed method. The efficiency of the classification is very high by presenting very good accuracy outcomes and also the classification of attacks is gives very accurate outcomes. From the outcomes, we have showed that the fuzzy classifier utilised in our proposed work outperforms the other classifiers ABC by facilitated very good accuracy. Thus, we can observe that our proposed work is better than other existing works for the attack classification.

References

Al-Maqaleh, B.M. and Shahbazkia, H. (2012) 'A genetic algorithm for discovering classification rules in data mining', *International Journal of Computer Applications*, Vol. 41, No. 18.

Ceravolo, P., Nocerino, M.C. and Viviani, M. (2004) 'Knowledge extraction from semi-structured data based on fuzzy techniques', *Knowledge-Based Intelligent Information and Engineering Systems*, Vol. 3215, pp.328–334.

Chia, H.W.K., Tan, C.L. and Sung, S.Y. (2006) 'Enhancing knowledge discovery via association-based evolution of neural logic networks', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, No. 7, pp.889–901.

Chow, M.Y., Altug, S. and Trussell, H.J. (1999) 'Heuristic constraints enforcement for training of and knowledge extraction from a fuzzy/neural architecture – part I: foundation', *IEEE Transactions on Fuzzy Systems*, Vol. 7, No. 2, pp.143–150.

Du, G., Jiang, Z., Diao, X. and Yao, Y. (2012) 'Knowledge extraction algorithm for variances handling of CP using integrated hybrid genetic double multi-group cooperative PSO and DPSO', *Journal of Medical Systems*, Vol. 36, No. 2, pp.979–994.

Fu, X. and Wang, L. (2001) 'Rule extraction by genetic algorithms based on a simplified RBF neural network', *Proceedings of the 2001 Congress on Evolutionary Computation*, Vol. 2, pp.753–758.

Gupta, M.K. and Sikka, G. (2013) 'Association rules extraction using multi-objective feature of genetic algorithm', *Proceedings of the World Congress on Engineering and Computer Science*, Vol. 2, San Francisco, USA.

Kadirkamanathan, V. and Fleming, P.J. (2005) 'Stability analysis of the particle dynamics in particle swarm optimizer', *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 3, pp.245–255.

Kumar, L. and Singh, D. (2013) 'Knowledge extraction from numerical data: an ABC based approach', *International Journal of Computer Engineering and Technology (IJCET)*, Vol. 4, No. 2, pp.1–9.

Liang, J.J., Qin, A.K. and Baskar, S. (2006) 'Comprehensive learning particle swarm optimizer for global optimization of multimodal functions', *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 3, pp.281–295.

Mittal, P. (2012) 'Knowledge extraction based on evolutionary learning (KEEL): analysis of development method, genetic fuzzy system', *International Journal of Computer Applications and Information Technology*, July, Vol. 1, No. 1, pp.1–4.

Nováček, V. (2008) 'Automatic knowledge acquisition and integration technique: application to large scale taxonomy extraction and document annotation', *Enterprise Information Systems*, Vol. 12, pp.160–172.

Papageorgiou, E.I. (2010) 'A novel approach on constructed dynamic fuzzy cognitive maps using fuzzified decision trees and knowledge-extraction techniques', *Studies in Fuzziness and Soft Computing*, Vol. 247, pp.43–70.

Pechenizkiy, M., Puuronen, S. and Tsymbal, A. (2003) 'Feature extraction for classification in knowledge discovery systems', *Knowledge-Based Intelligent Information and Engineering Systems*, Vol. 2773, pp.526–532.

Ratnaweera, A., Halgamuge, S.K. and Watson, H.C. (2004) 'Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients', *IEEE Transactions on Evolutionary Computation*, Vol. 8, No. 3, pp.240–255.

Rotshtein, A.P. and Mityushkin, Y.I. (2001) 'Extraction of fuzzy knowledge bases from experimental data by genetic algorithms', *Cybernetics and Systems Analysis*, Vol. 37, No. 4, pp.501–508.

Shanthi, S. and Bhaskaran, V.M. (2014) 'Modified artificial bee colony based feature selection: a new method in the application of mammogram image classification', *International Journal of Science, Engineering and Technology Research (IJSETR)*, Vol. 3, No. 6, pp.1664–1667.

Shivani, P., Dharmendrabhai and Gandhi, P. (2014) 'A detailed study on text mining using genetic algorithm', *International Journal of Engineering Development and Research (IJEDR)*, Vol. 1, No. 2, pp.108–113.

Shukran, M.A.M., Chung, Y.Y., Yeh, W.C., Wahid, N. and Zaidi, A.M.A. (2011) 'Artificial bee colony based data mining algorithms for classification tasks', *Modern Applied Science*, Vol. 5, No. 4, p.217.

Subanya, B. and Rajalaxmi, R.R. (2014) 'Artificial bee colony based feature selection for effective cardiovascular disease diagnosis', *International Journal of Scientific and Engineering Research*, Vol. 5, No. 5, pp.606–612.

Sumathi, T., Karthik, S. and Marikkannan, M. (2014) 'Artificial bee colony optimization for feature selection in opinion mining', *Journal of Theoretical and Applied Information Technology*, Vol. 66, No. 1.

Vivekanandan, P., Rajalakshmi, M. and Nedunchezhian, R. (2013) 'An intelligent genetic algorithm for mining classification rules in large datasets', *Computing and Informatics*, Vol. 32, pp.1–22.