REPUBLIC OF TURKEY
FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND
APPLIED SCIENCES

STUDENTS PERFORMANCE SYSTEM USING RECURRENT NEURAL NETWORK TRAINED BY MODIFIED GREY WOLF OPTIMIZER

DOSTI KHEDER ABBAS

Master Thesis
Department of Software Engineering
Supervisor: Assoc. Prof. Dr. Yalın Kılıç TÜREL

JULY – 2017

REPUBLIC OF TURKEY
FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

STUDENTS PERFORMANCE SYSTEM USING RECURRENT NEURAL NETWORK TRAINED BY MODIFIED GREY WOLF OPTIMIZER

MASTER THESIS

Dosti Kheder Abbas

151137112

Department: Software Engineering

Submission Date to the Institute: 20 June 2017

Thesis Presentation Date: 13 July 2017

Thesis Supervisor: Assoc. Prof. Dr. Yalın Kılıç TÜREL (F.U.)

Jury Member: Prof. Dr. Asaf VAROL (F.U.)

Jury Member: Assoc. Prof. Dr. Cihan VAROL (SHSU, USA) Cihan Varol

JULY – 2017

ACKNOWLEDGMENT

All praises to Allah for the strengths and his blessing in completing this work. I want to thank my supervisor Assoc. Prof. Dr. Yalin Kilic Turel. His office was always open to answer whenever I had a question about the research. Many thanks for Prof. Dr. Tarik Rashid for helping me in this thesis. Special thanks to my parents, sisters, and brothers.

Sincerely

Dosti Kheder Abbas

Elazig - 2017

TABLE OF CONTENTS

ABSTRACT

STUDENTS PERFORMANCE SYSTEM USING RECURRENT NEURAL NETWORK TRAINED BY MODIFIED GREY WOLF OPTIMIZER

A better quality of education can be obtained in educational institutions through a system that identifies the deficiencies of students and provides an initial warning to allow intervention for the students through counseling to help them in order to address their weaknesses. Current techniques need to be updated with such new soft computing approaches. The classification of the current techniques is not satisfactory. It needs to be studied with fresh techniques especially using hybrid techniques and those techniques which mimic mechanisms from nature. This research work aims at developing an intelligent approach using a modified Grey Wolf Optimizer (GWO) that mimics hunting style of grey wolves as an optimization algorithm with the Recurrent Neural Network (RNN) that mimics neurons of human's brain to forecast outcomes of students in a particular course based on their past achievements, social settings and the academic environments. It is a two-step procedure. Firstly, the Neural Network model is trained by using training dataset and its weights and biases are optimized through using the modified GWO. In the second step, to evaluate the trained model, the designed model is tested with a predefined testing dataset. For validation procedure, a 5-fold cross validation is used to obtain the best accuracy and performance. The results show that our approach obtains the best accuracy compared with certain other algorithms. This study can help an educational system to enhance students' learning experience as well as increasing their profits.

Keywords: Students' Performance, Classification, Recurrent Neural Network, Modified Grey Wolf Optimizer

ÖZET

DEĞİŞTİRİLEN BOZKURT OPTİMİZASYON YÖNTEMİ İLE EĞİTİLMİŞ YİNELENEN SİNİR AĞI KULLANAN ÖĞRENCİ PERFORMANS SİSTEMİ

Daha kaliteli bir eğitim öğrencilerin eksikliklerini belirleyen ve onlara zayıflıklarını göstermek için yardım etmek amacıyla rehberlik etmek yoluyla müdahale etmek için erken uyarı yapan bir system yoluyla eğitim kurumlarında elde edilebilir. Mevcut tekniklerin bazı yeni bilgisayar yazılım yaklaşımlarıyla güncellenmeye ihtiyaç duyar. Mevcut tekniklerin sınıflandırması memnun edici değildir. Bu sebeple, özellikle karma kullanılan yeni tekniklerle ve doğadaki bazı mekanizmaları taklit eden tekniklerle çalışmaya ihtiyaç duyulur. Bu araştırmada tekrarlı sinir ağı ile eniyileştirilmiş algoritma olarak bozkurtların avlanma tarzını taklit eden değiştirilmiş Bozkurt Eniyileştirmesi (GWO) kullanarak zeki bir yaklaşım geliştirilmesi amaçlanmıştır. Tekrarlı sinir ağı ise öğrencilerin sosyal çevrelerine, akademik ortamlarına ve geçmiş başarılarına dayalı özel bir kursta öğrencilerin gelecek başarılarını tahmin etmek için insan beynindeki nöronlarını taklit eder. Bu iki adımlı bir süreçtir. Öncelikle eğitilmiş bir veriseti kullarak sinir ağı modeli eğitilir ve çarpıklıklar değiştirilmiş GWO kullanarak eniyileştirilir. İkinci aşamada, eğitilen modeli değerlendirmek için tasarlanan model önceden belirlenen deneme veriseti ile test edilir. Doğrulama sürecinde en iyi doğruluğu ve performansı elde etmek için 5 katlı (5-fold) çapraz doğrulama kullanıldı. Sonuçlar bu yaklaşımın diğer mevcut algoritmalarla karşılaştırıldığında en doğru sonucun elde edilebileceğini göstermiştir. Bu çalışma öğrencilerin öğrenme deneyimlerini geliştirme ve daha verimli çalışma açısından eğitim kurumlarına yardımcı olabilir.

Anahtar Kelimeler: Öğrenci performansı, sınıflandırma, tekrarlı sinir ağları, değiştirilmiş bozkurt eniyileştirmesi

LIST OF FIGURES

Pg. No.

LIST OF TABLES

Pg. No.

ABBREVIATIONS

ACO:	Ant Colony Optimization
ANFIS:	Adaptive Neuro Fuzzy Inference System
ANN:	Artificial Neural Networks
BA:	Bees Algorithm
BPNN:	Back-Propagation Neural Network
CART:	Classification and Regression Tree
CGPA:	Cumulative Grade Point Average
CNN:	Cascading Neural Network
EDM:	Educational Data Mining
FFN:	Feedforward network
FN:	False Negative
FP:	False Positive
GPA:	Grade Point Average
GWO:	Grey Wolf Optimizer
MLP:	Multilayer Perceptron
MSE:	Mean Square Error
NPV:	Negative Predictive Value
PPV:	Positive Predictive Value
PSO:	Particle Swarm Optimization
RBF:	Radial Basis Function
RNN:	Recurrent Neural Network
SVM:	Support Vector Machine
TN:	True Negative
TNR:	True Negative Rate
TP:	True Positive
TPR:	True Positive Rate

1. INTRODUCTION

1.1. Overview

In educational management, students' performance perdition and classification is so significant. It would provide a suitable caution for students that did not perform well or those who's their performance is at risk and eventually assist the students to avert and overcome most of their issues in their missions. Yet, there are some challenges to gauge the students' performance as the academic performance of students is depending on various elements or features such as demographics, personal, educational background, psychological, academic progress and other environmental variables [1]. Statistical methods, Data mining and machine learning techniques are used for extracting useful information related to educational data, this is called Educational Data Mining (EDM) [2]. EDM uses academic databases and construct several techniques for identifying unique patterns [3, 4], to benefit academic planners in educational institutions through offering recommendations for improving the process of decision-making.

Academic performance research studies mostly have been carried out using classification and prediction methods. The task of classification is regarded as a process of determining a model in which data are classified to labels [5]. In the field of machine learning, neural networks are regarded as one of the best discoveries for classification problems that imitate neurons in human brain. The basic concepts of neural networks were proposed for the first time in 1943 [6]. Different kinds of neural networks were recommended in the literature such as the Feedforward Network (FFN) [7], kohonen self organizing network [8], Radial Basis Function (RBF) network [9], Recurrent Neural Network (RNN) [10], and spiking neural networks [11].

Neural networks are trained with backpropagation learning algorithm, they are usually slow, so it needs higher learning rate and momentum to get faster convergence. These approaches are very good only if the incremental training is required. However, they are still too slow for real life application. Nonetheless, Levenberg-Marquardt is used for small and medium size networks and it depends on the availability of memory, otherwise other fast algorithms would be alternatives. Backpropagation is deterministic algorithm which can tackle linearity and non-linearity problems. Yet, backpropagation and its variations may not always find a solution. Another problem associated with the back propagation algorithm is selecting learning rate. This is a complicated issue. For a liner network, a too large learning rate would cause unstable learning. Equally, a too small learning rate would cause longer time for training. The problem is more complex for nonlinear multilayer networks as it is hard to find an easy method for selecting a learning rate. The error surface for nonlinear networks is harder than that of liner network [12]. On the other hand, using neural networks with nonlinear transfer functions would present several local minima in the error surface. Thus, it is possible that a solution in a multi-layer network gets stuck in one of these local minima, this can be taken place depending on the initial starting conditions. It is worth mentioning that having a solution in local minim might be a satisfactory solution if the solution is close to the global minimum, otherwise the solution is bad. Another problem with the backpropagation is that it does not produce perfect weight connection for the best solution. In this case, the network needs to be reinitialized repeatedly to assurance the best solution [13, 14]. On contrary, Nature-Inspired Algorithms are stochastic, this means that the training session would start with random solutions and then they will get progressed. The essential element in a nature inspired algorithm is randomness. This means that the algorithms are using initial solutions and these solutions are improved in iterative style so that to avoid high local optima. Nature-inspired algorithms are considered to be useful for their simplicity, speed, faster convergence in finding a global optimum solution compared with deterministic methods. In addition, multilayer network is so subtle when it comes with hidden neurons in the hidden layer. The problem of under-fitting may arise when few neurons in the hidden layer is used, equally, over-fitting can arise when too many hidden neurons are used. RNNs are using less neurons in the hidden layer since there is a context layer, thus, less hidden neurons are needed and the networks is more stable and it can deal with temporal patterns.

In the EDM field, many classification techniques and algorithms were proposed. In this thesis, to assess students' outcomes in a specific module depending upon academic environments, students' achievements, and social settings, the RNN that imitates human brain's neurons with a modified Grey Wolf Optimizer (GWO) that imitates a hunting method of grey wolves as an optimization algorithm. The RNN model will be trained through a training dataset and the weights and biases of the neural network will be optimized using the modified GWO. Then, a testing dataset to evaluate the trained network will be used. To validate the procedure, 5-fold cross validation will be used to attain the highest performance and accuracy.

1.2. Literature Review

A neural network model [13] was used for forecasting students' performance in terms of Cumulative Grade Point Average (CGPA). The researchers used a data set that contains 120 records of students registered at Bangabandhu Sheikh Mujibur Rahman Science and Technology University. They used Levenberg Marquardt back propagation algorithm for training the neural network. The data set was divided into three sets; training, validating, and testing sets to make lower the error percentage. They concluded that early performance of students depends on academic and external activities such as living area condition, social media interaction, etc. It has been reported that neural networks have been successfully used for forecasting students' performance better than decision table, decision tree, and linear regression.

The ID3 classification method [14] was used for forecasting students' performance. The task was to extract information related to students' performance at the end of examination. This research study used data collected from Veer Bahadur Singh (VBS) Purvanchal University. Significant elements such as class test, attendance, assignment marks, and seminar were collected.

Student's actual mapping condition is a necessity that should be made before planning the performance enhancement program. A research study [15] has used K-means Clustering and focused on mapping students to uncover hidden patterns and grouping students dependent upon course attending average and their demographic such as gender, Grade Point Average (GPA), and certain courses' grade. Three clusters formed from three-hundred student's profiles namely: low-performance students, standard student and high-performance students.

The majority of students drop out of the courses are after the first year. Therefore, to predict the third-semester performance, in a research study [16] two classification techniques were used comparatively, namely J48 and Random Tree. The classification model was based on academic integration, social integration of student, and different unconsidered emotional skills. Researchers found that Random Tree is more accurate than the J48 algorithm in predicting performance.

Backpropagation neural network [17] has been used to predict student graduation outcomes. Development of such a network was presented as a three-layered perceptron and the backpropagation principles were used to train the network. Several experiments were applied in the training and testing phases. In these experiments, a set of 1,100 profiles were utilized to train the network and a set of 307 profiles were utilized for testing. The accuracy rate for the testing and training sets were 68% and 72%, respectively.

In the field of academic performance, there are some studies that predict lecturer's performance. A predicting Lecturers' performance study [18] has been studied to improve lecturers' performance and to enhance the accuracy of recognition system using the neural network combined with the Particle Swarm Optimization. Researchers evaluated lecturers based on three main criteria such as student feedback, continuous academic development, and lecturers' portfolio. Each of which contains a set of features. The three set of features combined and prepared as one dataset. Then the dataset has been preprocessed and important features are then fed as an input source to the training and testing phases. Particle Swarm Optimization has been used to find the best weights and biases in the training phase of the neural network. The best accuracy rate which obtained in the test phase was 98.28%.

A hybrid method [19] with the combination of decision tree and logistic regression was proposed to predict the students' dropout rate. Firstly, factors that influence dropouts are identified by using the decision tree algorithm J48. Then dropouts and the effect of each risk factor are quantified by applying logistic regression. In the study, students' cumulative grade point average GPA upon graduating has been used to evaluate the academic performance. Attendance, Gender, previous semester grade, parent incomes, parent education, first child, student working, and scholarship are risk factors that the relationship between underlying them and students dropouts has been studied to start the modeling process.

A Dropouts' prediction study [20] was examined in an online program. Researchers obtained a dataset consists 189 students through online questionnaires. The main subjects of the questionnaires were demographic survey, readiness for online learning questionnaire, online technologies self-Efficacy scale, prior knowledge questionnaire, and locus of control scale. Four classification algorithms were applied to classify students' dropout named decision tree, nearest neighbour, neural network, and naive bayes. 10-fold cross validation was used for the training and testing of the algorithms. The accuracy rate of the decision tree was 79.7%, nearest neighbour was 87%, neural network was 76.8%, and naive bayes was 73.9%. The study found that the most significant factors in forecasting the dropouts were online learning readiness, online technologies self-efficacy, and previous online experience.

Grouping students is a common task in education hence many researchers have used data mining techniques for this task. Researchers [21] used data mining applications to profile and group students. To profile students, they made use of one of the common approaches to mine associations that discovers co-relations between a set of objects which is Apriori algorithm so that students can easily be grouped and invisible patterns of their learning style are identified. Moreover, undesirable behaviors of students can be found. For mining highest association rules, they have used Weka tool to extract information dependent upon which they can profile performance of the students as Poor, Satisfactory or Good. Attendance, term work marks, practical exams and exam scores are the various parameters that are used to apply both of the algorithms to group and profile students. An efficient way for profiling students was proposed by the application of the algorithms which can be used in educational systems.

Association rule mining was applied [22] using Apriori algorithm to enhance the quality and predict performances of students in university result. Researchers have obtained student's data set from Pune University that pursues master of computer application degree. The analysis showed that the performance of university students is based on the unit test, attendance, assignment, and graduation percentage. In this study, various association rules have been found between attributes. Moreover, it is found that how the attributes influence the result of university students. Results showed that the features unit test, assignment, attendance, and graduation are significant for identifying students who are poor. This can enhance the performance level of students and gives them further guidance to enhance the university result.

1.3. Problem Statement

Although researchers have presented various approaches for improving the education quality by increasing students' performance but for taking into consideration some challenges are still exist in the field of predicting students' performance. The thesis locates tiring issue as well as yielding to lack of accuracy in evaluation process, which is one of the issues to evaluate prediction of students' performance if it is to be done manually. Moreover, the evaluation tips to time consuming. The outcomes from the approaches in the literature are not capable of being satisfied while most of the studies suggested appropriate methods to evaluate the students' performance. Back Propagation Neural Networks (BPNN) are usually slow, so it needs higher learning rate and momentum to get faster convergence. These approaches are very good only if the incremental training is required. Another problem with the backpropagation is that it does not produce perfect weight connection for the best solution.

1.4. Aim of the Thesis

The main objectives of this research study are to develop an intelligent, computerized, dynamic, time efficient, accurate system that enhances the education quality in Middle East academic institutions, specifically in Iraq. We aim to demonstrate how student's past achievements, social settings, and the academic environments influence student's outcome and performance in a course. Nature-inspired algorithms are considered to be useful for their simplicity, speed, faster convergence in finding a global optimum solution compared with deterministic methods. In this thesis a modified GWO is used to optimize weights and biases of RNN. This technique is used to see how effective is it? in training the RNN compared with BPNN and other types of neural networks. Moreover, this thesis provides extensive conclusions and suggestions for the future research work or for other applications and making a bridge between the previous studies and the future studies.

1.5. Contributions

In this thesis, the contributions in the area of predicting students' performance center on the reality of the data which was collected from Salahaddin University. Additionally, a computerized method is implemented in order to improve this area in the university via the features that the university identified to examine the students' performance. A modified GWO with RNN is proposed for the first time in the field of students' performance that makes the thesis near optimum.

1.6. Thesis Layout

The rest of the thesis is organized as follows:

Chapter Two: [Students' Performance Prediction System]

This chapter includes background information about students' performance prediction, popular algorithms for performance evaluation, standard students' performance system phases and algorithms utilized in each phase.

Chapter Three: [Proposed Approach]

The proposed approach for prediction of students' performance is described in this chapter. It shows the structure of the system, the student datasets, data preprocessing, feature selection, and classification.

Chapter Four: [Experimental Results]

The results of using the classification algorithms are presented in this chapter

Chapter Five: [Conclusions and Recommendations]

This chapter includes the discussion and summarization of the final results of the work done. Recommendations for future work are presented to improve the system.

2. STUDENTS’ PERFORMANCE SYSTEM USING ANALYSIS AND ALGORITHMS

2.1. Introduction

In this chapter, background information about analyzing and understanding the students’ performance system are presented. Some artificial intelligence approaches and algorithms that are utilized for assessing and classifying students’ performance or outcomes are discussed and explained. The general and standard form of performance measurement process is explained in the following discussion.

2.2. Standard Model of the General Performance Measurement Process

The process of general performance measurement is structured in the five following steps: targeting, indicator selection, data collection, analysis, and reporting. Figure 2.1 shows the five phases below [23].

Figure 2.1 Standard model of the procedure of general performance measurement

2.2.1. Targeting

In performance measurement, the starting phase is targeting to try to find what to measure. A clear picture of the organization or the institution should be available to answer the question of what to measure to find out the requirements of targeting. In this phase, there are three issues to be highlighted. Firstly, in order to discover what to measure, the map of the institution should be known. This procedure will be implemented by representing the institution or a sector through management models, organizational charts, stakeholders’ analysis, programme theory and programme logic. Secondly, to explain which section of the institution should be studied, the targets of measurement effort should be known. The third issue to be highlighted is clarifying the argumentation for selecting a goal, which goes deeper into the sector to see the features of the selected section [23].

2.2.2. Indicator Selection

After selecting what part to measure, we need to know how to measure this part. This is done via the second phase in performance measurement which is selecting indicators. Specialized experts in the sector are able to select indicators. Good indicators are provided via several criteria such as understandability for users, precise identification, documented, timely, relevant, and feasible [24].

2.2.3. Data Collection

Data collection is the third phase in performance measurement. In this phase, one of the important decisions is whether the institution uses external or internal data sources. External data is a data source that is obtained or purchased outside the institution or from other institutions while the internal data source obtained inside the institution. Internal data is usually cheaper than external data source. When the internal data is not used or not sufficient some other data sources are considered such as extra data registration, self-assessments, survey, technical measurement, registrations of other institutions, and statistical institutions [24].

2.2.4. Analysis

Analysis is the fourth phase that demonstrates numbers. The aim of analysis is to convert or transform unclear data to information that can be helpful in the decision-making process. Norm, breakouts and causal analysis are the three interpretive strategies of analysis. Norm aims at providing references for example, time-dimension, scientists and political foundations, and comparison with other institutions. The breakout aims at breaking the data to parts in order to understand where, when and for whom the data is related. The causal strategy works on finding the cause of under performance of the selected feature [25].

2.2.5. Reporting

Reporting is the last phase of performance measurement. The main goal of reporting is to provide a suitable format for the target group. In the reporting phase, two questions should be answered. The first is who is consuming the information? Is it the Head of Department, Dean, University Rector, Secondly, what is a suitable format? Is it in the form of an annual report, an annual plan or financial documents [25]?

2.3. Standard Stages of Students' Performance Prediction

Standard phases of the predicting performance system consist of five phases such as input data phase, preprocessing phase, feature selection phase, classification phase, and results evaluation. At each phase, distinctive approaches have been applied. In the discussion below, the approaches of each phase are discussed briefly.

2.3.1. Input Data Stage

A student performance system has various systems in the literature depending on different features. Some of the studies worked on predicting academic outcomes. In addition, grouping students and relation between dependent and independent variables were studied. In general, to collect input data, different methods have been used such as organizational data [17], questionnaires [26], expert knowledge and literature review [27] etc.

Although, there are public datasets related to students' performance prediction available, not all features may be beneficial for improving the quality of education and students' performance system in Iraq, because it needs an applicable procedure to select features agreeable with the model on hand. Also, we want to test the designed model with a real dataset. As a compromise, we decided to use previously collected data from Salahaddin University. This was a good decision as we did not need to collect the data again. Also, it allowed more time to work on the implementation of the algorithms. Furthermore, the system can also be used in Salahaddin University and for other universities or organizations that use similar features not only in Iraq even in neighboring countries.

2.3.2. Preprocessing Stage

Data preprocessing is an important phase in the prediction of students' performance system as it prepares the data to be utilized in feature selection and classification phases. Misleading results might be produced if the data was not collected carefully. Therefore, the data quality is primary before running the other phases. The training and other phases will be more difficult to process if the data is noisy, irrelevant, and redundant.

An imbalanced dataset is one of the major problems that acts as a hindrance for generalization. A dataset is imbalanced if classification labels are not equally distributed. In this thesis we have distributed our dataset over five sets for cross validation with an equal number of data samples in regard of class labels.

2.3.3. Feature Selection Stage

The most noteworthy phase is the feature selection phase in creating a suitable prediction system for students' performance. The approach of feature selection makes the data ready to be utilized in the next phase. For the performance prediction system, much data can be obtained and collected, but some of this data can be eliminated since it may not have a great effect to a good prediction. Thus, we should filter and reduce the obtained data to a smaller dataset that could give a better accuracy level. If the dataset has features with significant effects on the model, then the structured model will have a better accuracy.

Different filters and algorithms have been joined to obtain better feature selection accuracy. In this thesis, feature selection using Correlation Attribute Evaluation in Weka tool will be employed. This method examines the value of attributes through measuring the correlation between them and the class.

Correlation Attribute Evaluation is a feature selection method that filters based on the correlation between features with the class label. Nominal features are evaluated on a value-by-value basis through treating the values as indicators. A weighted average is obtained which is an overall correlation for a nominal attribute.

2.3.4. Classification Stage

After the feature selection phase, the classification phase comes. Features or attributes will be fed to the classification algorithm to take the role of input data, which forecasts a performance category beforehand. Various methodologies have been proposed in the literature for performance classification. The most popular algorithms used for this intent are Fuzzy Logic Classifier [28, 29], Adaptive Neuro Fuzzy Inference System (ANFIS) [27], Naive Bayes classification [30], Support Vector Machine (SVM) [31], and BPNN [30, 25]. There are little studies on RNN with natured inspired optimization algorithms specifically GWO. Thus, in this thesis, we will add this combination with modification in the algorithms. Three types of neural network algorithms are applied in this thesis for classifying students' performance named Multilayer Perceptron (MLP), Cascading Neural Network (CNN), and Recurrent Neural Network (RNN). In addition, GWO with a modification is used as an optimization algorithm. These algorithms are explained below.

2.3.4.1. Neural Network

In the area of Computational Intelligence, the significant developments are the Neural Networks. Neural networks imitate the brain of human neurons mostly to solve classification task. The general idea of neural networks was firstly suggested in 1943. Various neural network kinds were proposed in the literature for example Feedforward network, Kohonen self-organizing network, Radial basis function network, Recurrent Neural Network, and Spiking Neural Networks.

Neural network types are the same in learning. The capability of a neural network in learning from experience is referred to be learning. Like the neurons in humans' brains, the Artificial Neural Networks (ANN) have been provided with techniques to adjust themselves to a set of given inputs. Two common kinds supervised [32, 33] and unsupervised [34, 35] are available. If the neural network is equipped from an external source with feedbacks, then it is a supervisor. In contrast, in unsupervised learning, a neural network adjusts itself to inputs without any extra external feedbacks [36].

In general, a trainer provides learning for a neural network. The trainer can be considered as the most significant element of any neural network. A trainer is in control of training neural networks to obtain the highest performance in supervised learning. Firstly, a training method equips neural networks with a set of samples called training samples. Then, to enhance the performance, the trainer updates the structural parameters of the neural network in each training stage. The trainer is omitted once the training stage is completed and the neural network will be ready to use.

In this thesis, we have used three types of neural network named RNN, MLP, and CNN as classifiers. But, we have proposed and focused on RNN as classifier because it provides higher accuracy than the other neural networks. That is why in our methodology we choose RRN combined with a modified GWO to classify students.

2.3.4.1.1. Multilayer Perceptron (MLP)

A Feed-forward Neural Network (FNN) with one hidden layer is called multilayer perceptron [36]. FNN is a neural network with only one-directional connections between it's neurons. In a FNN, nodes are organized in different layers [7]. The first layer is called the input layer. The output layer is defined as the last layer. The layer(s) between the output and input layers is/are defined to be the hidden layer(s). A structural design of MLP is shown in Figure 2.2 [36].

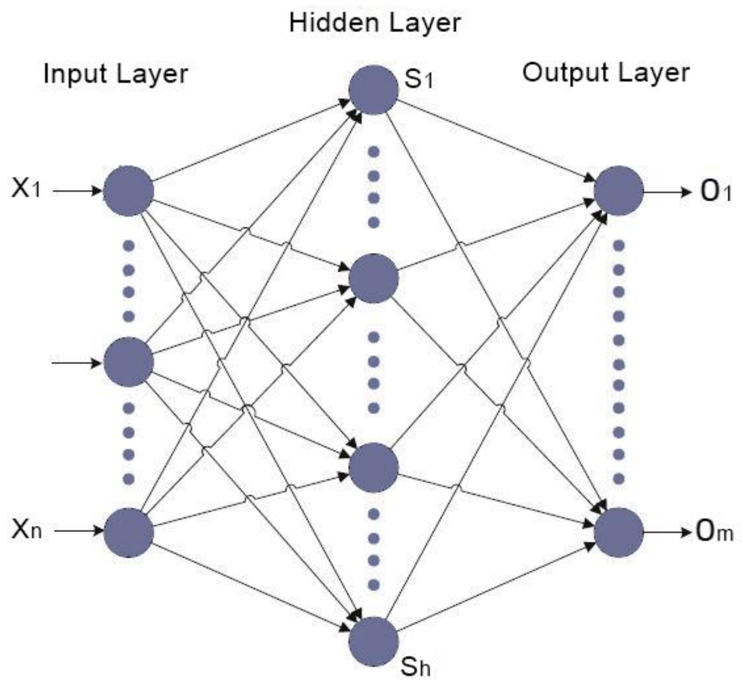

Figure 2.2 MLP (One Hidden Layer)

The output of a MLP is calculated after equipping the inputs, weights, and biases by the following equations [37].

Firstly, the weighted sums of inputs are calculated by equation 1.

$$S_j = \sum_{i=1}^{n} W_{ij}.X_i - \theta_j , \qquad j = 1,2, \dots , h \tag{1}$$

Where n indicates the input nodes number, W_{ij} is the connection weight from the i^{th} node in the input layer to the j^{th} node in the hidden layer, θ_j shows the bias of the j^{th} hidden node, and X_i is the i^{th} input.

The calculation of the output of hidden nodes is done by using activation functions. In this thesis, we use sigmoid functions to calculate the output of nodes in each layer as follows.

$$S_j = sigmoid(S_j) = \frac{1}{(1 + e^{-S_j})}, \qquad j = 1,2, \dots, h \tag{2}$$

After the outputs of hidden nodes are defined then based on their output, the final outputs will be defined as follows:

$$O_k = \sum_{j=1}^{h} W_{jk}.S_j - \theta'_k, \quad k = 1,2, \dots, h \tag{3}$$

$$O_k = sigmoid\ (O_k) = \frac{1}{(1 + exp^{-O_k})}, \quad k = 1,2, \dots, h \tag{4}$$

Where W_{jk} indicates the connection weight from the j_{th} hidden node to the k_{th} output node, and θ'_k indicates the bias of the k_{th} output node.

As can be seen in equation 3 and 4, the final output of MLPs is defined by the weights and biases from given inputs. The perfect definition of MLPs' training is obtaining recommendable relation between the inputs and outputs and finding appropriate values for weights and biases.

2.3.4.1.2. Recurrent Neural Network (RNN)

Conventional feedforward neural networks spread data flow linearly from an input layer to subsequent layers. RNNs are considered as bi-directional data flow neural networks. The data flow propagates from previous processing phases to earlier phases. In this thesis, the basic idea of a simple RNN is used which was firstly proposed by Jeff Elman [38]. This kind of RNN is held a simple modulation in the design of the basic FFN.

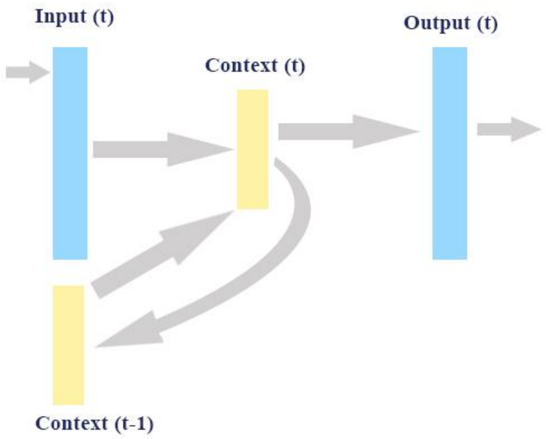

Figure 2.3 A simple RNN Model

As it is demonstrated in Figure 2.3, a three-layered network was used by the RNN. The output at time $(t-1)$ is held from the hidden layer then inserted to the input layer as a context. After that the units in the context with input nodes are fed back at time (t) to the hidden layer. With that manner, context nodes always maintain a duplicated value of the previous hidden nodes because of the propagation through the recurrent connections at time$(t-1)$, before a parameter-updating rule is applied at time (t). Accordingly, the network design can learn through a set of state summarizing previous inputs. Dynamics of this type of neural network is represented in the below equation [39]:

$$h(t) = f\big(Ux(t) + Vh(t-1)\big) \tag{5}$$

Where U and V are weight matrices between the initial input and the hidden nodes, and between the hidden nodes and the context nodes, respectively while W indicates the output weight matrix.

After finding the value of $h(t)$ from equation 5, the output of nodes in the hidden and the output layers is calculated similarly as its calculated in the FNN and MLPs. Firstly, the weighted sums of inputs are calculated by equation 1. The calculation of the output of hidden nodes is done by using sigmoid function using equation 2. After the outputs of hidden nodes are defined then based on their output, the final outputs will be defined using equation 3 and 4.

2.3.4.1.3. Cascading Neural Network (CNN)

Cascade-Correlation or Cascading Neural Network (CNN) is a supervised learning algorithm in neural networks which was proposed in 1990 [40] for the first time. This type of neural network starts training with a minimal network instead of only adjusting weights in a network of fixed architecture, and then new hidden units will be automatically trained and added one by one to create a multi-layer model. Whenever a new hidden node has been inserted to the network then its input-side weights are frozen.

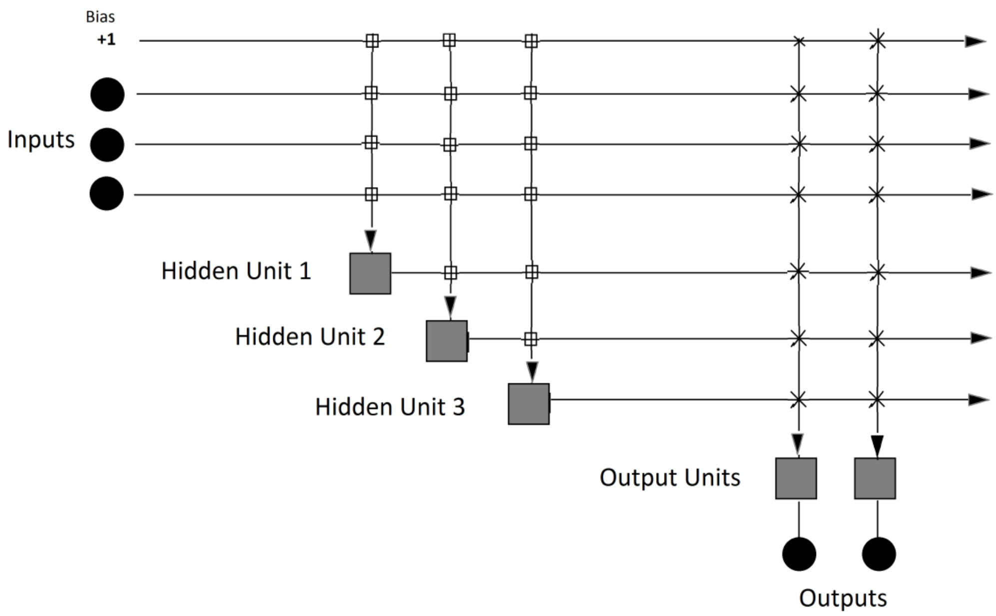

Figure 2.4 The CNN architecture, after three hidden units have been added (Adapted from [40])

Figure 2.4 illustrates the structure of the CNN. The vertical lines indicate summing all incoming activation. Boxed shows frozen connections, *X* indicates connections are trained repeatedly. The starting point of the network is some input nodes and one or more output nodes with no hidden nodes. The I/O representation and the problem dictate the number of inputs and outputs. All the input units are connected to all the output units through a connection with an adaptable weight. Also a bias input is available [40].

Hidden nodes will be added one by one to the network. Each of the original input units of the network sends a connection to each new hidden node and also the new hidden node receives connection from the entire pre-existing hidden node. Whenever the node is added to the net, the input weights of the hidden unit are frozen while just the output connections are trained iteratively. A very strong high-order feature retriever will be created since new nodes add a new one-node (layer) to the network, except if some of its entering weights give a result of zero. Moreover, very deep networks might be led by this process.

The output of nodes in the hidden and the output layers is calculated similarly as its calculated in the FNN and MLPs. Firstly, the weighted sums of inputs are calculated by equation 1. The calculation of the output of hidden nodes is done by using sigmoid function by equation 2. After the outputs of hidden nodes are defined then based on their output, the final outputs will be defined using equation 3 and 4.

Two key ideas can be combined by cascading neural network. The first one is in the cascade architecture. One hidden node is added to the network at a time and after it has been added, it is not changed. The second idea is the learning algorithm that produces and installs new hidden nodes. In this type of neural network, the magnitude of the correlation between the output of the new node and the residual error signal will be maximized.

2.3.4.2. Grey Wolf Optimizer (GWO)

One of the swarm-based metaheuristic algorithms is the Grey Wolf Optimizer. Most of the meta-heuristic algorithms are nature-inspired that imitate a technique in nature to provide optimization tasks for example Bees Algorithm (BA) [41], Ant Colony Optimization (ACO) [42], and Particle Swarm Optimization (PSO) [43]. Some of the algorithms were imitated from behaviors of animals such as the ACO algorithm, which was inspired by ant colonies behavior. BA, which was inspired by behavior of honey bees in food foraging, and the GWO, which was inspired by the grey wolves' hunting style. Generally, maximizing neural network's performance is the main purpose of utilizing meta-heuristic algorithms combined with neural networks. Moreover, they are considered to be useful for their simplicity, speed, faster convergence in finding a global optimum solution compared with deterministic methods.

The GWO is a fresh algorithm, which was proposed by Mirjalili [44]. The population in this algorithm is distributed into four groups such as alpha(α), beta(β), delta(δ), and omega(ω). (α), (β), and (δ) are indicated as the first three best wolves in the population. These wolves will guide (ω) wolves to find suitable areas of the search space. Equation 6 and 7 are used to find Omegas positions during optimization around(α), (β), and (δ) as follows [44]:

$$\vec{D} = \left| \vec{C} \cdot \vec{X}p(t) - \vec{X}(t) \right| \tag{6}$$

$$\vec{X}(t+1) = \vec{X}p(t) - \vec{A} \cdot \vec{D} \tag{7}$$

Where $\vec{D}$ is distance of wolves toward the prey, t denotes the current iteration, $\vec{X}p$ indicates position vector of the prey, $\vec{X}$ is the position vector of a grey wolf, $\vec{A} = 2a \cdot \vec{r}1a$, $\vec{C} = 2 \cdot \vec{r}2$, $\vec{r}1$ and $\vec{r}2$ are random vectors in [0,1], and a is linearly decreased from 2 to 0.

The thought behind updating position is demonstrated in Figure 2.5 through equations 6 and 7. It can be observed in the equations and the figure that a wolf can update her location in the position of(X, Y) around the prey.

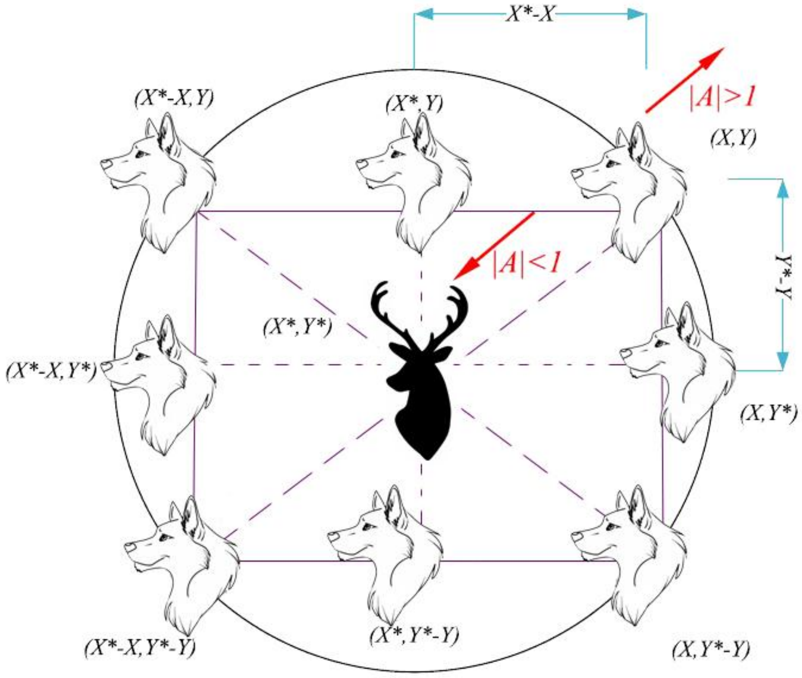

Figure 2.5 The technique of position updating of search agents and impacts of $\boldsymbol{A}$ on it (Adapted from [44])

In the algorithm, the optimum or the prey position is supposed to be (α), (β), and (δ) all the time. Therefore, the best three solutions during optimization are set as(α), (β), and (δ) respectively. The other wolves (ω)s are capable of changing their position with respect to (α), (β), and (δ). Calculations of the approximate distance between (α), (β), and (δ) and the current solution are demonstrated in the following equations [44]:

$$\vec{D}\alpha = |\vec{C}1 \cdot \vec{X}\alpha - \vec{X}| \tag{8}$$

$$\vec{D}\beta = |\vec{C}2 \cdot \vec{X}\beta - \vec{X}| \tag{9}$$

$$\vec{D}\delta = |\vec{C}3 \cdot \vec{X}\delta - \vec{X}| \tag{10}$$

Where $\vec{X}\alpha$, $\vec{X}\beta$, and $\vec{X}\delta$ are the position of alpha, beta, and delta, $\vec{X}$ is the position of the current solution, and $\vec{C}1$, $\vec{C}2$, and $\vec{C}3$ are random vectors.

The step size of omega wolves will be defined toward alpha, beta and delta respectively by the equations 8, 9 and 10.

After defining the distances, the calculations of the final position for the current solution are demonstrated below [44]:

$$\vec{X}1 = \vec{X}\alpha - \vec{A}1 \cdot (\vec{D}\alpha) \tag{11}$$

$$\vec{X}2 = \vec{X}\beta - \vec{A}2 \cdot (\vec{D}\beta) \tag{12}$$

$$\vec{X}3 = \vec{X}\delta - \vec{A}3 \cdot (\vec{D}\delta) \tag{13}$$

$$\vec{X}(t+1) = \frac{\vec{X}1 + \vec{X}2 + \vec{X}3}{3} \tag{14}$$

Where $\vec{A}1$, $\vec{A}2$, and $\vec{A}3$ denote random vectors, and t is the iterations' number.

In the GWO algorithm, $\vec{A}$ and $\vec{C}$, which are random and adaptive vectors are used for providing both exploration and exploitation as it is demonstrated in figure 2. As can be seen the exploration occurs if $|A| > 1$ or $|A| < -1$. The exploration is also helped by the vector $\vec{C}$ if it is bigger than 1. However, if $\vec{A}$ is smaller than 1, and $\vec{C}$ is smaller than 1, then the exploitation occurs. A suitable technique is suggested in the algorithm to solve the entrapment of local optima. Thus, to emphasize exploitation, it is noticed during optimization that as the iteration counter increases A gets decreased linearly. Yet, C is randomly produced during the optimization to emphasize exploration or exploitation at any stage.

The pseudo code of the GWO algorithm is demonstrated below [44]:

$Initialize\ the\ grey\ wolf\ population\ Xi(i = 1,2,\dots,n)$
$Initialize\ a, A, and\ C$
$Calculate\ the\ fitness\ of\ each\ search\ agent$
$X\alpha = the\ best\ search\ agent$
$X\beta = the\ second\ best\ search\ agent$
$X\delta = the\ third\ best\ search\ agent$
$\boldsymbol{while}\ (t < Max\ number\ of\ iterations)$
$\boldsymbol{for}\ each\ search\ agent$
$Update\ the\ position\ of\ the\ current\ search\ agent\ by\ equation\ (14)$
$\boldsymbol{end\ for}$
$Update\ a, A, and\ C$
$Calculate\ the\ fitness\ for\ all\ search\ agents$
$Update\ X\alpha, X\beta, and\ X\delta$
$t = t + 1$
$\boldsymbol{end\ while}$
$return\ X\alpha$

2.3.5. Results and Evaluation

After the data is classified, the results are evaluated based on the targets of the research. In this thesis, besides evaluating the results based on the research goals we also evaluate and validate the results using a confusion matrix. A confusion matrix is a table that is mostly used to show the classification performance of a model. The table consists of True Positive (TP), False Negative (FN), True Negative (TN), and False Positive (FP) values, which are used to calculate some cases. TP reports true positive result in the testing data, FP reports error in the positive values. TN reports true negative test result in the data, FN reports error in the negative values. The table of confusion matrix is shown below:

Table 2.1 Confusion matrix

	Predicted Condition	
	Predicted Condition Positive	**Predicted Condition Negative**
Actual Condition	TP	FN
	FP	TN

These values can be found in confusion matrix such as sensitivity or True Positive Rate (TPR), specificity or True Negative Rate (TNR), the Positive Predictive Value (PPV) or precision which determines the success rate in positive samples, and Negative Predictive Value (NPV) which indicates the success rate in negative samples. Additionally, the accuracy will be found. The mentioned values are calculated based on the equations which are introduced below [45, 46]:

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{15}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{16}$$

$$\text{PPV} = \frac{TP}{TP + FP} \tag{17}$$

$$\text{NPV} = \frac{TN}{TN + FN} \tag{18}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{19}$$

3. PROPOSED APPROACH FOR STUDENTS' PERFORMANCE SYSTEM

3.1. Introduction

We discussed the theory of predicting students' performance in the previous chapter. Various approaches and algorithms that have been utilized in the literature in the performance prediction area were presented. In this thesis, we have used three types of neural network named RNN, MLP, and CNN as classifiers. But, we have proposed and focused on RNN as classifier as it provides higher accuracy than the other neural networks. That is why in our methodology we choose RRN combined with a modified GWO to classify students. In this chapter, we will explain the structure of the proposed approach for the students' performance prediction system. Firstly, the architectural design of the system is described. After that, we describe the other phases of the implementation of the system.

3.2. Structure of the Proposed Technique

There should be a structure that manages our study before applying approaches of the data mining on the dataset. Figure 3.1 illustrates the methodology used in this thesis. In general, the problem is defined firstly, then the data will be preprocessed, after that, we apply the process of feature selection. In the classification stage, RNN is used as classifier with the modified GWO to optimize weights and biases of the neural network. Finally the results and evaluation is coming. We discuss the structure of the system in the following sections.

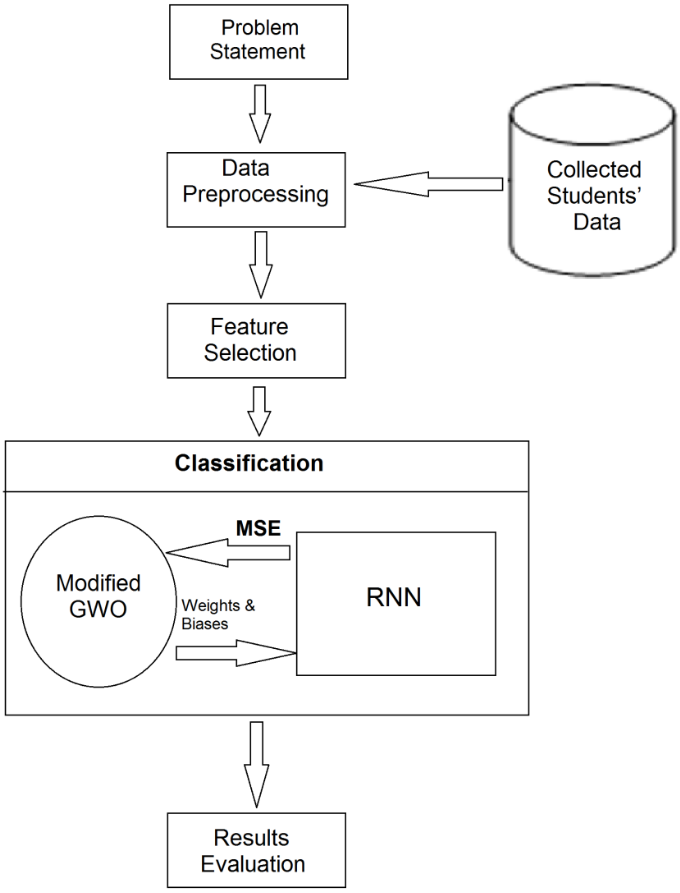

Figure 3.1 Structure of the proposed technique

3.2.1. Experimental Setup

Initially, data preprocessing is coming to modify the data which was collected from Salahaddin University to a format suitable for data preprocessing. The dataset descriptions are described in section 3.2.2. The next phase is preprocessing to apply balancing of the data in each of the classes. Then, the dataset samples are divided into five similar sized sets for the aim of using (5)-folds cross validation. Feature selection is then performed. The proposed classification technique developed a modified GWO to optimize weights and biases of a RNN model. Theories of the above mentioned classifier have been discussed in the previous chapter. Details of the model stages will be discussed in the following sections.

3.2.2. The Student Dataset

Selecting a suitable dataset is a challenging problem as there are many data resources but most of them may not be suitable for every model. In this thesis use a real dataset which was collected by Rashid and his colleague [47] was arranged beforehand to consist of samples with detail of various features which affect performance of student. In this thesis, we consider the first year students from the departments of electrical engineering, civil engineering, software engineering, architectural engineering, mechanical engineering, survey and dam and water resources engineering, and geology sciences. The dataset is collected for the academic year (2012-2013). In this thesis, this data will be used as our dataset to perform the proposed technique for the students' classification. The dataset information consists of academic environments, social settings, and students past achievements so that the research is mostly based on tutors expertise and the socio-economic background.

The features of the dataset are described in table 3.1. In this table, it is described that each of the features has been set with a specific number of variables. The data is set with real numbers as it is designed in the table. For example, 0 to 5 was set the values of father's education level feature, which means, value 0 assigns PhD, 1 assigns MSc/MA, 2 assigns, 3 assigns BSc/BA, 4 assigns diploma, and 5 assigns none.

Table 3.1 The student dataset descriptions

No	Features	Description
1	Gender	Female (154 students), Male (133 students)
2	Age	Contains three values
3	Address (Town)	İt is distributed into three zones (Students who located so far to the city that the university locates in, Students who located far to the city, and Students who located close to or in the city)
4	Address (City)	İt is distributed into three zones (Students who located far to the city that the university locates in, Students who located close to the city, and Students who located in the city)
5	Level of Mother's Education	PhD, MSc/MA, BSc/BA, Diploma, None
6	Level of Father's Education	PhD, MSc/MA, BSc/BA, Diploma, None
7	High School (HS) Address (Village)	It is distributed into three zones (HSs so far to the city that the university locates in, HSs far to the city, and HSs close to or in the city)
8	HS Address (Town)	It is distributedin to three zones (HSs so far to the city that the university locates in, HSs far to the city, and HSs close to or in the city)
9	HS Address (City)	It is distributed into three zones (HSs far to the city that the university locates in, HSs close to the city, and HSs in the city)
10	HS Language	English, Kurdish, Arabic, Other languages
11	HS Type	Private, Public
12	English module Score at national exam	Excellent, Very Good, Good, Medium, Pass, Fail
13	Overall score at national exam (7 subjects)	Excellent, Very Good, Good, Medium, Pass, Fail
14	Department	Electrical engineering, Civil engineering, Mechanical engineering, Software engineering, Survey and Dam and water Resources Engineering, Architectural engineering, and Geology Sciences
15	College	Engineering, Science
16	The Course Test score –General University Exam	Excellent, Very Good, Good, Medium, Pass, Fail
17	The Course Test score –Department Exam	Excellent, Very Good, Good, Medium, Pass, Fail
18 - 20	English Tutor (Contains three features: English local, Internal local, and native)	Yes, No
21	Class	Failed, Passed

3.2.3. Data Preprocessing

Students past achievements, social settings and the academic environments are the main features in our dataset to be retrieved correctly. A suitable format to the data should be provided for the classification task after the data was collected on student's records and questionnaires. After that, the data is normalized due to the processing. The data is then balanced manually dependent on the output class. As the cross validation is used, the dataset will be arranged in a structure, which consists of (5)-folds. Each fold has similar number of failed and passed students in compare with the other folds.

3.2.4. Feature Selection

After data preprocessing, the feature selection process is coming. The most related attributes to the class labels will be selected. Determining the features makes the network have a higher accuracy rate because it eliminates the features with less relation to the performance. In this thesis, those features that have not any impact on the output are removed. Moreover, only one attribute will be chosen from those that have similarity in affecting the output. In this step, Correlation Attribute Evaluation is used in Weka tool. This method evaluates the worth of features by measuring the correlation between them and the class. The weight results of the evaluation using Correlation Attribute Evaluation for the feature selection are shown in Table 3.2.

Table 3.2 Weight of the Features

No	Feature	Weight
1	Module Test – General University Exam	0.3964
2	English Score at National Exam	0.3911
3	Overall Score at National Exam	0.2762
4	Department	0.2495
5	Tutor Native	0.16
6	Mother's Education Level	0.1527
7	High School Language	0.1483
8	Father's Education Level	0.1407
9	Module Test – Department Exam	0.1296
10	Tutor English Local	0.1241
11	Address of High School (Village)	0.1239
12	Address of High School (Town)	0.1239
13	Personal Address (Town)	0.1048
14	Age	0.0948
15	Address of High School (City)	0.094
16	High School Type	0.0736
17	Tutor Internal Local	0.0636
18	Personal Address (City)	0.0602
19	Gender	0.0553
20	College	0

It is observed in the table that the weight value of attribute College is zero, which means it has not any impact on the output value. Moreover, High School Address for the Village and Town had similar weight value, which means they have the same impact on the output value. Therefore, High School (Village) and College features are removed. Before performing the classification task, the dataset is consisted of eighteen input attributes and one output attribute.

3.2.5. Classification

There are several conventional classification algorithms in the education data mining filed. In this thesis, a modified GWO combined with RNN will be utilized to classify students' outcomes in a module. This process falls into two steps. At the first, the RNN model will be trained through a training dataset. The RNN's weights and biases will be optimized by using the modified GWO. Secondly, the designed RNN is tested by a previously defined testing dataset for evaluation of the trained model. (5) Folds cross validation is used to validate the procedure for obtaining the highest performance and accuracy.

3.2.5.1. Modified Grey Wolf Optimizer

In In this thesis, a variant of GWO is produced through adding two simple modifications in the original GWO algorithm to optimize parameters of the recurrent neural network to classify students. The outcomes demonstrate that the modifications have affected the classification accuracy positively. As mentioned in the above, the population in GWO algorithm is divided into four sets: Alpha (α), Beta (β), Delta (δ), and Omega (ω). Alpha, Beta, and Delta are recognized as the first three fittest wolves or the best solutions that direct the Omega wolves on the way to achieve capable search space area. The first modification, is basically adding another best solution along with Alpha, Beta, and Delta called Gamma (see equation 20). When the Omega wolves update their positions with respect to the best positions, this time they respectively update their positions with more best positions (Alpha, Beta, Delta and Gamma) than the standard algorithm.

The second modification is proposed when the step size of omega wolves (which moves on the way to Alpha, Beta, Delta and Gamma respectively) is defined as demonstrated in equations11, 12, 13, and 14. Whenever $\vec{X}1$, $\vec{X}2, \vec{X}3$, and $\vec{X}4$ are calculated, instead of utilizing the distances of Alpha, Beta, Delta, and Gamma ($\vec{D}\alpha, \vec{D}\beta, \vec{D}\delta,$ and $\vec{D}\gamma,$) individually, the average of them will be used in the equations. Updated and inserted equations are demonstrated below:

$\vec{D}\alpha, \vec{D}\beta,$ and $\vec{D}\delta$ are found by equations (8, 9, and 10)

$$\vec{D}\gamma = \left| \vec{C}4 \cdot \vec{X}\gamma - \vec{X} \right| \tag{20}$$

Where $\vec{D}\gamma$ is the approximate distance between Gamma and the current solution. $\vec{X}\gamma$ displays the position of Gamma, $\vec{X}$ is the position of the current solution, and $\vec{C}4$ is a random vector. The value of $\vec{C}$ was defined above in the GWO Algorithm.

$$\vec{D}avg = \frac{\vec{D}\alpha + \vec{D}\beta + \vec{D}\delta + \vec{D}\gamma}{4} \tag{21}$$

Where $\vec{D}avg$ denotes the average of the approximate distances between Alpha, Beta, Delta, and Gamma and the current solution respectively.

Then, equations 11, 12, and 13 will be updated as follows:

$$\vec{X}1 = \vec{X}\alpha - \vec{A}1 \cdot (\vec{D}avg) \tag{22}$$

$$\vec{X}2 = \vec{X}\beta - \vec{A}2 \cdot (\vec{D}avg) \tag{23}$$

$$\vec{X}3 = \vec{X}\delta - \vec{A}3 \cdot (\vec{D}avg) \tag{24}$$

Where $\vec{A}1$, $\vec{A}2$, and $\vec{A}3$ denote random vectors. The value of $\vec{A}$ was defined above in the GWO Algorithm.

Moreover, before calculating the final position of the current solution another equation is inserted as follow:

$$\vec{X}4 = \vec{X}\gamma - \vec{A}4 \cdot (\vec{D}avg) \tag{25}$$

Where $\vec{A}4$ denotes a random vector.

At the end, to calculate the final position of the current solution, the equation (14) is updated as follow:

$$\vec{X}(t+1) = \frac{\vec{X}1 + \vec{X}2 + \vec{X}3 + \vec{X}4}{4} \tag{26}$$

Where t is the number of iterations.

The pseudo code of the modified GWO is demonstrated below. The differences between the standard GWO and the modified GWO are differed by ←.

$Initialize\ the\ grey\ wolf\ population\ Xi(i = 1,2, \dots, n)$
$Initialize\ a, A, and\ C$
$Calculate\ the\ fitness\ of\ each\ search\ agent$
$X\alpha = the\ best\ search\ agent$
$X\beta = the\ second\ best\ search\ agent$
$X\delta = the\ third\ best\ search\ agent$
$X\gamma = the\ fourth\ best\ search\ agent$ ←
$Calculate\ the\ average\ distance\ between\ the\ current\ search\ agent$
$and\ X\alpha, X\beta, X\delta, and\ X\gamma$ ←
$\boldsymbol{while}\ (t < Max\ number\ of\ iterations)$
$\boldsymbol{for}\ each\ search\ agent$
$Update\ the\ position\ of\ the\ current\ search\ agent\ by\ equation\ (26)$ ←
$\boldsymbol{end\ for}$
$Update\ a, A, and\ C$
$Calculate\ the\ fitness\ for\ all\ search\ agents$
$Update\ X\alpha, X\beta, X\delta, and\ X\gamma$ ←
$t = t + 1$
$\boldsymbol{end\ while}$
$return\ X\alpha$

3.2.5.2. Trained RNN through the modified GWO

In this thesis, (5)-folds cross validation was used to verify the classification task. Figure 3.2 demonstrates the training phase, which will be processed in each folds execution.

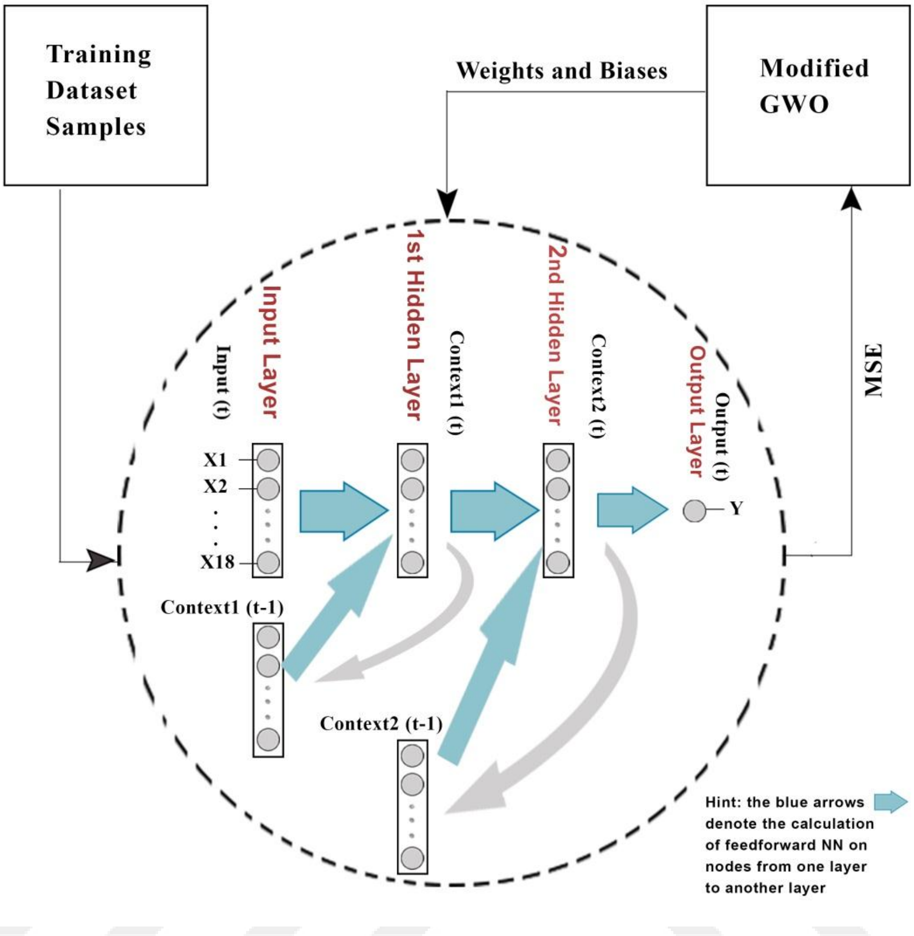

Figure 3.2 Trained RNN through the modified GWO

Here, it should be observed that the concept of the simple type of RNN is applied on the proposed neural network, which consists of a multilayer perceptron with two hidden layers. The proposed neural network starts with 18 nodes in the input layer, 10 nodes in the first hidden layer (Context1), 10 nodes in the second hidden layer Context2, and one node in the output layer. Moreover, to perform the process of RNN, two groups of the context nodes that contain a copy of Context1 in the input layer at time $(t-1)$ and a copy of Context2 in the first hidden layer at time $(t-1)$. Also, it should be noted that Mean Square Error (MSE) is the input to the modified GWO and weights and biases are the output. The objective function to train the neural network is obtaining the least MSE to attain the best classification accuracy. MSE is defined as the difference between the obtained and the desirable value in the RNN. MSE can be calculated as the equation below:

$$MSE = \frac{1}{n}\sum_{j=1}^{n}(O_j^k - d_j^k)^2 \quad (27)$$

Where n is the outputs number. When the K^{th} training sample is used, d_j^k denotes the desired output of j^{th} input node and O_j^k denotes the actual output of j^{th} input node.

At the first in training stage, the modified GWO starts with initializing its variables and weights and biases in form of vectors. Each value in the vector is assigned with a weight or a bias in the RNN. After that, first sample will feed to the network. The output at time $(t-1)$ in the first hidden layer (Context1) is held and will be inserted to the input layer as a group, which is called context units. The output at time $(t-1)$ (Context2) in the second hidden layer is held and inserted to the first hidden layer. Then, context1 will feed back to the first hidden layer at time(t) with the initial inputs and context2 will feed back to the second hidden layer with the current Context1. The designed neural network can learn and preserve from a set of state summarizing previous inputs. Calculations of the proposed RNN are always done by equation 5 in terms of summing the context with the input or hidden nodes, and the sigmoid function is used acting as the activation function.

After finding the value of $h(t)$ from equation 5, the output of nodes in the hidden and the output layers is calculated similarly as its calculated in the FNN and MLPs. Firstly, the weighted sums of inputs are calculated by equation 1. The calculation of the output of hidden nodes is done by using sigmoid function by equation 2. After the outputs of hidden nodes are defined then based on their output, the final outputs will be defined using equation 3 and 4.

Iteratively, the same initialized weights and biases are used to feed the other training samples to the RNN. The proposed RNN has been significantly effective as the whole set of training samples are adapted by the RNN. The modified GWO will receive the MSE after calculating the MSE over the whole training samples. The modified GWO compares the MSE with fitness amongst the four best wolves (solutions), which are (α), (β), (δ), and (γ). Then, the position and the fitness of each of the best wolves will be updated. Weights and bias vectors that assign the search agents positions are updated iteratively depending on the search agent's number with respect to $(\alpha), (\beta)$, (δ), and (γ). After updating weights and biases, the modified GWO will send the weight and biases to the RNN. Once again, the updated weights and biases and the training samples are utilized for training the RNN to attain a new MSE. This procedure will be continued until the last iteration. At the end, the optimized weights and biases can be used to test the network through testing dataset without utilizing the modified GWO.

4. COMPUTER SIMULATION TESTS AND RESULTS

4.1. Introduction

One of the important phases in this research study is to test the system and record the experimental results. Various cases are studied to choose the most accurate and suitable possible results. The variety of results gives the possibility of selecting an accurate algorithm for the proposed system besides identifying the cause of getting low accuracy results. The features of the dataset that are used as input data, the classifier parameters, the output class label, and the results of both training and testing phase of the classifier will be described in this chapter.

4.2. Experimental Results Using Modified GWO with RNN

Table 4.1 demonstrates the results of the classification using cross validation for modified GWO with RNN. We have divided the dataset into five sets (5)-folds. The sets were denoted as X1, X2, X3, X4, and X5. Number of samples in each group is similar or close to each other. The first three sets which are X1, X2, and X3 contain 57 samples but X4 and X5 consist of 58 samples. Four folds will be fed to the neural network in each execution. These folds are consisted of 230 samples and they will act as the training dataset. The other fold that contains 57 samples will act as the testing dataset for testing the neural network.

The results demonstrate the training rates of the folds as 99.5%, 99.5%, 99.5%, 99.1, and 99.5 resulting in an average training rate of 99.4%. Moreover, the testing rates of the classification in the folds are demonstrated as 96.4%, 100.0%, 100.0%, 98.2%, and 98.2 resulting in an average testing rate of 98.6%. In the results, it is observed that a better classification rate is provided by a smaller MSE. For example, MSE for testing phase is 0.009 in Fold (1) and the testing classification rate is 96.4%, but a smaller MSE 0.002 in the second or the third fold has provided 100% testing classification rate.

Table 4.1 Classification results for the modified GWO with RNN

Fold No.	Training/ Testing	Dataset	No. Samples	MSE	Classification Rate
Fold (1)	**Training**	X2+X3+X4+X5	230	0.0029211	99.56%
	Testing	X1	57	0.0091276	96.49%
Fold (2)	**Training**	X1+X3+X4+X5	230	0.002923	99.56%
	Testing	X2	57	0.0027236	100.00%
Fold (3)	**Training**	X1+X2+X4+X5	230	0.0030488	99.56%
	Testing	X3	57	0.0027124	100.00%
Fold (4)	**Training**	X1+X2+X3+X5	229	0.0030902	99.12%
	Testing	X4	58	0.0030253	98.27%
Fold (5)	**Training**	X1+X2+X3+X4	229	0.0028664	99.56%
	Testing	X5	58	0.0033664	98.27%
Average	**Training**			0.0029699	99.47%
	Testing			0.00419106	98.60%

Table 4.2 demonstrates students' classification results in the testing phase. It is observed in the table that number of students is totally 287. Number of the students who are passed in the module is totally 183 and number of the students who are failed in the course is totally 104. 36 Students out of 37 passed students were correctly classified in the first run with 97.29% success rate. Contrary in the same run, 19 students out of 20 failed students were correctly classified with 95.00% success rate. Results are similar in the second and the third fold. All the samples were correctly classified from the passed and failed students with 100.00% success rate. Once more, results are similar in Fold (4) and Fold (5). 35 Students out of 36 passed students were correctly classified with 97.22% success rate. In contrast, all the students were correctly classified with 100.00% success rate.

In general, 180 students out of 183 passed students were classified successfully with 98.34% success rate. In contrast, 103 students out of 104 failed students were correctly classified with 99.00% success rate.

Table 4.2 Classification performance and the students' outcomes in the modified GWO with RNN

Fold No.	Data set	No. Samples	Passed Students			Failed Students		
			No. Students	**No. Correctly Classified Students**	**Success Rate**	**No. Students**	**No. Correctly Classified Students**	**Success Rate**
Fold (1)	X1	57	37	36	97.29%	20	19	95.00%
Fold (2)	X2	57	37	37	100.00%	20	20	100.00%
Fold (3)	X3	57	37	37	100.00%	20	20	100.00%
Fold (4)	X4	58	36	35	97.22%	22	22	100.00%
Fold (5)	X5	58	36	35	97.22%	22	22	100.00%
Total		287	183	180		104	103	
Aver-age					98.34%			99.00%

In this thesis, we use confusion matrix to evaluate the students' classification results in the proposed classification techniques. The testing results for the modified GWO with RNN are evaluated by using confusion matrix in the following discussion.

Table 4.3 displays the confusion matrix in the first fold for the proposed classification algorithm. The predicted number of true positives (passed) was 36, the predicted number of false negatives (Failed) was 1, the predicted number of false positives (passed) was 1, and the predicted number of true negatives (failed) was 19.

Table 4.3 Confusion matrix for modified GWO with RNN – Fold (1)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	36	1
	Failed	1	19

From the table, we can find sensitivity or True Positive Rate (TPR), specificity or True Negative Rate (TNR), the Positive Predictive Value (PPV) or precision which determines the success rate in passed students, and Negative Predictive Value (NPV) which determines the success rate in failed students. Moreover, the accuracy of the network will be found. The mentioned variables are calculated based on the equations which were introduced in chapter two as shown below:

$$\text{Sensitivity} = \frac{36}{36+1} = 0.97\,, \qquad \text{Specificity} = \frac{19}{19+1} = 0.95$$

$$\text{PPV} = \frac{36}{36+1} = 0.97\,, \qquad \text{NPV} = \frac{19}{19+1} = 0.95$$

$$\text{Accuracy} = \frac{36+19}{36+19+1+1} = 0.96$$

From the calculations, we can see that the value of sensitivity was 0.97 meaning that TPR was 97%. The value of specificity was 0.95 meaning that TNR was 95%, the value of PPV was 0.97 meaning that the success rate in passed students was 97%, the value of NPV was 0.95 meaning that the success rate in failed students was 95%, and the obtained accuracy of the network in the first fold was 96%.

Table 4.4 displays the confusion matrix in the second fold for the proposed classification algorithm. The predicted number of true positives was 37, the predicted number of false negatives was 0, the predicted number of false positives was 0, and the predicted number of true negatives was 20.

Table 4.4 Confusion matrix for modified GWO with RNN – Fold (2)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	37	0
	Failed	0	20

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{37}{37+0} = 1.00\,, \qquad \text{Specificity} = \frac{20}{20+0} = 1.00$$

$$\text{PPV} = \frac{37}{37+0} = 1.00\,, \qquad \text{NPV} = \frac{20}{20+0} = 1.00$$

$$\text{Accuracy} = \frac{37+20}{37+20+0+0} = 1.00$$

From the calculations, the value of sensitivity was 1.00 meaning that TPR was 100%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that the success rate in passed students was 100%, the value of NPV was 1.00 meaning that success rate in failed students was 100%, and the obtained accuracy of the network in the second fold was 100%.

Table 4.5 displays the confusion matrix in the third fold for the proposed classification algorithm. The predicted number of true positives was 37, the predicted number of false negatives was 0, the predicted number of false positives was 0, and the predicted number of true negatives was 20.

Table 4.5 Confusion matrix for modified GWO with RNN – Fold (3)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	37	0
	Failed	0	20

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{37}{37+0} = 1.00\,, \qquad \text{Specificity} = \frac{20}{20+0} = 1.00$$

$$\text{PPV} = \frac{37}{37+0} = 1.00\,, \qquad \text{NPV} = \frac{20}{20+0} = 1.00$$

$$\text{Accuracy} = \frac{37+20}{37+20+0+0} = 1.00$$

From the calculations, we can see that the value of sensitivity was 1.00 meaning that TPR was 100%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that success rate in passed students was 100%, the value of NPV was 1.00 meaning that success rate in failed students was 100%, and the obtained accuracy of the network in the third fold was 100%.

Table 4.6 shows the confusion matrix in the fourth fold for the proposed classification algorithm. The predicted number of true positives was 35, the predicted number of false negatives was 0, the predicted number of false positives was 1, and the predicted number of true negatives was 22.

Table 4.6 Confusion matrix for modified GWO with RNN – Fold (4)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	35	0
	Failed	1	22

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{35}{35+0} = 1.00\,, \qquad \text{Specificity} = \frac{22}{22+1} = 0.95$$

$$\text{PPV} = \frac{35}{35+1} = 0.97\,, \qquad \text{NPV} = \frac{22}{22+0} = 1.00$$

$$\text{Accuracy} = \frac{35+22}{35+22+1+0} = 0.98$$

From the calculations, we can see that the value of sensitivity was 1.00 meaning that TPR was 100%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that success rate in passed students was 100%, the value of NPV was 1.00 meaning that success rate in failed students was 100%, and the obtained accuracy of the network in the fourth fold was 98%.

Table 4.7 shows the confusion matrix in the fifth fold for the proposed classification algorithm. The predicted number of true positives was 35, the predicted number of false negatives was 0, the predicted number of false positives was 1, and the predicted number of true negatives was 22.

Table 4.7 Confusion matrix for modified GWO with RNN – Fold (5)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	35	0
	Failed	1	22

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{35}{35+0} = 1.00\,, \qquad \text{Specificity} = \frac{22}{22+1} = 0.95$$

$$\text{PPV} = \frac{35}{35+1} = 0.97\,, \qquad \text{NPV} = \frac{22}{22+0} = 1.00$$

$$\text{Accuracy} = \frac{35+22}{35+22+1+0} = 0.98$$

From the calculations, we can see that the value of sensitivity was 1.00 meaning that TPR was 100%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that success rate in passed students was 100%, the value of NPV was 1.00 meaning that success rate in failed students was 100%, and the obtained accuracy of the network in the fifth fold was 98%.

4.3. Experimental Results Using Standard GWO with RNN

The results of the classification using cross validation for standard GWO with RNN are displayed in table 4.8. The same structure for dividing data samples over the folds as modified GWO with RNN is used. It is observed in the results that the classification rates in training phase for the folds were 99.5%, 98.6%, 94.3%, 99.5%, and 99.5% resulting in an average training rate of 98.3%. Moreover, the testing classification rates for the folds were observed as 98.24%, 98.24%, 82.45%, 98.27%, and 94.82% resulting in an average testing rate of 94.4%.

Table 4.8 Classification results for the GWO with RNN

Fold No.	Training/ Testing	Dataset	No. Samples	MSE	Classification Rate
Fold (1)	**Training**	X2+X3+X4+X5	230	0.0028909	99.56%
	Testing	X1	57	0.0031643	98.24%
Fold (2)	**Training**	X1+X3+X4+X5	230	0.0029296	98.69%
	Testing	X2	57	0.0034252	98.24%
Fold (3)	**Training**	X1+X2+X4+X5	230	0.0030763	94.34%
	Testing	X3	57	0.0056983	82.45%
Fold (4)	**Training**	X1+X2+X3+X5	229	0.0030833	99.56%
	Testing	X4	58	0.0032921	98.27%
Fold (5)	**Training**	X1+X2+X3+X4	229	0.0026807	99.56%
	Testing	X5	58	0.0032239	94.82%
Average	**Training**			0.002932	98.34%
	Testing			0.003761	94.40%

Students' classification results for testing phase are presented in table 4.9. It is observed in the table that the results in the first and the second fold are the same. In the mentioned folds, all the passed students which are 37 students were correctly classified with 100.00% success rate. Also, 19 students out of 20 failed students were correctly classified with 95.00% success rate. In the third run, 36 students out of 37 passed students were correctly classified with 97.29% success rate. In contrast 11 students out of 20 failed students were correctly classified with 55.00% success rate. In fold 4, all the passed students which are 36 were correctly classified with 100.00% success rate. In contrast, 21 students out of 22 failed students were correctly classified with 95.45% success rate. In the fifth fold, 34 students out of 36 passed students were correctly classified with 94.44%

success rate. In contrast, 21 samples out of 22 failed students were correctly classified with 95.45% success rate.

In general, 180 students out of 183 passed students were classified successfully with 98.35% success rate. In contrast, 91 samples out of 104 failed students were correctly classified with 87.18% success rate.

Table 4.9 Classification performance and the students' outcomes in the GWO with RNN

			Passed Students			Failed Students		
Fold No.	**Data set**	**No. Samples**	**No. Students**	**No. Correctly Classified Students**	**Success Rate**	**No. Students**	**No. Correctly Classified Students**	**Success Rate**
Fold (1)	X1	57	37	37	100.00%	20	19	95.00%
Fold (2)	X2	57	37	37	100.00%	20	19	95.00%
Fold (3)	X3	57	37	36	97.29%	20	11	55.00%
Fold (4)	X4	58	36	36	100.00%	22	21	95.45%
Fold (5)	X5	58	36	34	94.44%	22	21	95.45%
Total		287	183	180		104	91	
Aver-age					98.35%			87.18%

The testing results for the GWO with RNN are evaluated by a confusion matrix in the following discussion.

Table 4.10 displays the confusion matrix in the first fold for the proposed classification algorithm. The predicted number of true positives was 37, the predicted number of false negatives was 1, the predicted number of false positives was 0, and the predicted number of true negatives was 19.

Table 4.10 Confusion matrix for GWO with RNN – Fold (1)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	37	1
	Failed	0	19

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{37}{37+1} = 0.97\,, \qquad \text{Specificity} = \frac{19}{19+0} = 1.00$$

$$\text{PPV} = \frac{37}{37+0} = 1.00\,, \qquad \text{NPV} = \frac{19}{19+1} = 0.95$$

$$\text{Accuracy} = \frac{37+19}{37+19+0+1} = 0.98$$

From the calculations, we can see that the value of sensitivity was 0.97 meaning that TPR was 97%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that success rate in passed students was 100%, the value of NPV was 0.95 meaning that success rate in failed students was 95%, and the obtained accuracy of the network in the first fold was 98%.

Table 4.11 shows the confusion matrix in the second fold for the proposed classification algorithm. The predicted number of true positives was 37, the predicted number of false negatives was 1, the predicted number of false positives was 0, and the predicted number of true negatives was 19.

Table 4.11 Confusion matrix for GWO with RNN – Fold (2)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	37	1
	Failed	0	19

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{37}{37+1} = 0.97\ , \qquad \text{Specificity} = \frac{19}{19+0} = 1.00$$

$$\text{PPV} = \frac{37}{37+0} = 1.00\ , \qquad \text{NPV} = \frac{19}{19+1} = 0.95$$

$$\text{Accuracy} = \frac{37+19}{37+19+0+1} = 0.98$$

From the calculations, we can see that the value of sensitivity was 0.97 meaning that TPR was 97%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that success rate in passed students was 100%, the value of NPV was 0.95 meaning that success rate in failed students was 95%, and the obtained accuracy of the network in the second fold was 98%.

Table 4.12 shows the confusion matrix in the third fold for the proposed classification algorithm. The predicted number of true positives was 36, the predicted number of false negatives was 9, the predicted number of false positives was 1, and the predicted number of true negatives was 11.

Table 4.12 Confusion matrix for GWO with RNN – Fold (3)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	36	9
	Failed	1	11

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{36}{36+9} = 0.80\,, \qquad \text{Specificity} = \frac{11}{11+1} = 0.91$$

$$\text{PPV} = \frac{36}{36+1} = 0.97\,, \qquad \text{NPV} = \frac{11}{11+9} = 0.55$$

$$\text{Accuracy} = \frac{36+11}{36+11+1+9} = 0.82$$

From the calculations, we can see that the value of sensitivity was 0.80 meaning that TPR was 80%. The value of specificity was 0.91 meaning that TNR was 91%, the value of PPV was 0.97 meaning that success rate in passed students was 97%, the value of NPV was 0.55 meaning that success rate in failed students was 55%, and the obtained accuracy of the network in the third fold was 82%.

Table 4.13 shows the confusion matrix in the fourth fold for the proposed classification algorithm. The predicted number of true positives was 36, the predicted number of false negatives was 1, the predicted number of false positives was 0, and the predicted number of true negatives was 21.

Table 4.13 Confusion matrix for GWO with RNN – Fold (4)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	36	1
	Failed	0	21

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{36}{36+1} = 0.97\,, \qquad \text{Specificity} = \frac{21}{21+0} = 1.00$$

$$\text{PPV} = \frac{36}{36+0} = 1.00\,, \qquad \text{NPV} = \frac{21}{21+1} = 0.95$$

$$\text{Accuracy} = \frac{36+21}{36+21+0+1} = 0.98$$

From the calculations, we can see that the value of sensitivity was 0.97 meaning that TPR was 97%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that success rate in passed students was 100%, the value of NPV was 0.95 meaning that success rate in failed students was 95%, and the obtained accuracy of the network in the fourth fold was 98%.

Table 4.14 shows the confusion matrix in the fifth fold for the proposed classification algorithm. The predicted number of true positives was 34, the predicted number of false negatives was 1, the predicted number of false positives was 2, and the predicted number of true negatives was 21.

Table 4.14 Confusion matrix for GWO with RNN – Fold (5)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	34	1
	Failed	2	21

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{34}{34+1} = 0.97\,, \qquad \text{Specificity} = \frac{21}{21+2} = 0.91$$

$$\text{PPV} = \frac{34}{34+2} = 0.94\,, \qquad \text{NPV} = \frac{21}{21+1} = 0.95$$

$$\text{Accuracy} = \frac{34+21}{34+21+2+1} = 0.94$$

From the calculations, we can see that the value of sensitivity was 0.97 that means TPR was 97%. The value of specificity was 0.91 that means TNR was 91%, the value of PPV was 0.94 that means success rate in passed students was 94%, the value of NPV was 0.95 that means success rate in failed students was 95%, and the obtained accuracy of the network in the fifth fold was 94%.

4.4. Experimental Results Using Modified GWO with CNN

The results of the classification using cross validation for modified GWO with RNN are shown in table 4.15. The same structure of dividing data samples over the folds as modified GWO with RNN is used. Results showed that the training classification rates in the folds were 92.17%, 90.00%, 83.91%, 92.57%, and 91.70% resulting in the average of 90.07%. Also, the classification rates for the testing phase for each fold were 92.98%, 80.70%, 82.45%, 91.37%, and 84.48 resulting in the average of 86.40%.

Table 4.15 Classification results for the modified GWO with CNN

Fold No.	Training/ Testing	Dataset	No. Samples	MSE	Classification Rate
Fold (1)	**Training**	X2+X3+X4+X5	230	0.0024027	92.17%
	Testing	X1	57	0.002574	92.98%
Fold (2)	**Training**	X1+X3+X4+X5	230	0.0029054	90.00%
	Testing	X2	57	0.0040148	80.70%
Fold (3)	**Training**	X1+X2+X4+X5	230	0.0057163	83.91%
	Testing	X3	57	0.0057401	82.45%
Fold (4)	**Training**	X1+X2+X3+X5	229	0.0026008	92.57%
	Testing	X4	58	0.0027498	91.37%
Fold (5)	**Training**	X1+X2+X3+X4	229	0.0026523	91.70%
	Testing	X5	58	0.0029477	84.48%
Average	**Training**			0.0032555	90.07%
	Testing			0.003605	86.40%

Table 4.16 shows students' classification results for the testing phase. It is observed in the table that all the samples out of 37 passed students were correctly classified in the first execution with 100.00% success rate. Contrary in the same execution, 16 samples out of 20 failed students were correctly classified with 80.00% success rate. In fold (2), 36 samples out of 37 passed students were correctly classified with 97.29% success rate. In contrast, half of the samples out of 20 failed students were correctly classified with 50.00% success rate. In fold (3), 34 students out of 37 passed students were correctly classified with 91.89% success rate. In contrast, 13 students out of 20 failed students were correctly classified with 65.00% success rate. In fold (4), 34 students out of 36 passed students were correctly classified with 94.44% success rate. 19 Students out of 22 failed students were correctly classified with 86.36% success rate. In the last execution, 34 students out of 36 passed students were correctly classified with 94.44% success rate. In contrast, 15 samples out of 22 failed students were correctly classified with 68.18% success rate.

In general, 175 students out of 183 passed students were classified successfully with 95.61% success rate. In contrast, 73 samples out of 104 failed students were correctly classified with 69.91% success rate.

Table 4.16 Classification performance and the students' outcomes in the modified GWO with CNN

Fold No.	Data set	No. Samples	Passed Students			Failed Students		
			No. Students	**No. Correctly Classified Students**	**Success Rate**	**No. Students**	**No. Correctly Classified Students**	**Success Rate**
Fold (1)	X1	57	37	37	100.00%	20	16	80.00%
Fold (2)	X2	57	37	36	97.29%	20	10	50.00%
Fold (3)	X3	57	37	34	91.89%	20	13	65.00%
Fold (4)	X4	58	36	34	94.44%	22	19	86.36%
Fold (5)	X5	58	36	34	94.44%	22	15	68.18%
Total		287	183	175		104	73	
Aver-age					95.61%			69.91%

The testing results for the GWO with RNN are evaluated by confusion matrix in the following discussion.

Table 4.17 shows the confusion matrix in the first fold for the proposed classification algorithm. The predicted number of true positives was 37, the predicted number of false negatives was 4, the predicted number of false positives was 0, and the predicted number of true negatives was 16.

Table 4.17 Confusion matrix for the modified GWO with CNN – Fold (1)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	37	4
	Failed	0	16

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{37}{37+4} = 0.90\,, \qquad \text{Specificity} = \frac{16}{16+0} = 1.00$$

$$\text{PPV} = \frac{37}{37+0} = 1.00\,, \qquad \text{NPV} = \frac{16}{16+4} = 0.80$$

$$\text{Accuracy} = \frac{37+16}{37+16+0+4} = 0.89$$

From the calculations, we can see that the value of sensitivity was 0.90 meaning that TPR was 90%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 1.00 meaning that success rate in passed students was 100%, the value of NPV was 0.80 meaning that success rate in failed students was 80%, and the obtained accuracy of the network in the first fold was 89%.

Table 4.18 shows the confusion matrix in the second fold for the proposed classification algorithm. The predicted number of true positives was 36, the predicted number of false negatives was 10, the predicted number of false positives was 1, and the predicted number of true negatives was 10.

Table 4.18 Confusion matrix for the modified GWO with CNN – Fold (2)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	36	10
	Failed	1	10

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{36}{36+10} = 0.78\,, \qquad \text{Specificity} = \frac{10}{10+0} = 1.00$$

$$\text{PPV} = \frac{36}{36+1} = 0.97\,, \qquad \text{NPV} = \frac{10}{10+10} = 0.50$$

$$\text{Accuracy} = \frac{36+10}{36+10+1+10} = 0.80$$

From the calculations, we can see that the value of sensitivity was 0.78 meaning that TPR was 78%. The value of specificity was 1.00 meaning that TNR was 100%, the value of PPV was 0.97 meaning that success rate in passed students was 97%, the value of NPV was 0.50 meaning that success rate in failed students was 50%, and the obtained accuracy of the network in the second fold was 80%.

Table 4.19 shows the confusion matrix in the third fold for the proposed classification algorithm. The predicted number of true positives was 34, the predicted number of false negatives was 7, the predicted number of false positives was 3, and the predicted number of true negatives was 13.

Table 4.19 Confusion matrix for the modified GWO with CNN – Fold (3)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	34	7
	Failed	3	13

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{34}{34+7} = 0.82\,, \qquad \text{Specificity} = \frac{13}{13+3} = 0.81$$

$$\text{PPV} = \frac{34}{34+3} = 0.91\,, \qquad \text{NPV} = \frac{13}{13+7} = 0.65$$

$$\text{Accuracy} = \frac{34+13}{34+13+3+7} = 0.82$$

From the calculations, we can see that the value of sensitivity was 0.82 meaning that TPR was 82%. The value of specificity was 0.81 meaning that TNR was 81%, the value of PPV was 0.91 meaning that success rate in passed students was 91%, the value of NPV was 0.65 meaning that success rate in failed students was 65%, and the obtained accuracy of the network in the third fold was 82%.

Table 4.20 shows the confusion matrix in the fourth fold for the proposed classification algorithm. The predicted number of true positives was 34, the predicted number of false negatives was 3, the predicted number of false positives was 2, and the predicted number of true negatives was 19.

Table 4.20 Confusion matrix for the modified GWO with CNN – Fold (4)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	34	3
	Failed	2	19

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are shown below:

$$\text{Sensitivity} = \frac{34}{34+3} = 0.91\,, \qquad \text{Specificity} = \frac{19}{19+2} = 0.90$$

$$\text{PPV} = \frac{34}{34+2} = 0.94\,, \qquad \text{NPV} = \frac{19}{19+3} = 0.86$$

$$\text{Accuracy} = \frac{34+19}{34+19+2+3} = 0.91$$

From the calculations, we can see that the value of sensitivity was 0.91 meaning that TPR was 91%. The value of specificity was 0.90 meaning that TNR was 90%, the value of PPV was 0.94 meaning that success rate in passed students was 94%, the value of NPV was 0.86 meaning that success rate in failed students was 86%, and the obtained accuracy of the network in the fourth fold was 91%.

Table 4.21 shows the confusion matrix in the fifth fold for the proposed classification algorithm. The predicted number of true positives was 34, the predicted number of false negatives was 7, the predicted number of false positives was 2, and the predicted number of true negatives was 15.

Table 4.21 Confusion matrix for the modified GWO with CNN – Fold (5)

		Predicted	
		Passed	**Failed**
Actual class	**Passed**	34	7
	Failed	2	15

The calculations of sensitivity, specificity, PPV, NPV, and the accuracy are calculated as shown below:

$$\text{Sensitivity} = \frac{34}{34+7} = 0.82\,, \qquad \text{Specificity} = \frac{15}{15+2} = 0.88$$

$$\text{PPV} = \frac{34}{34+2} = 0.94\,, \qquad \text{NPV} = \frac{15}{15+7} = 0.68$$

$$\text{Accuracy} = \frac{34+15}{34+15+2+7} = 0.84$$

From the calculations, we can see that the value of sensitivity was 0.82 meaning that TPR was 82%. The value of specificity was 0.88 meaning that TNR was 88%, the value of PPV was 0.94 meaning that success rate in passed students was 94%, the value of NPV was 0.68 meaning that success rate in failed students was 68%, and the accuracy of the network in the fifth fold was determined as 84%.

4.5. Comparing the Proposed Technique with Other Techniques

Figure 4.1 displays the performance of our method the modified GWO with RNN which acquired the highest accuracy among some techniques. For evaluation, the technique was compared to some techniques. As it is observed in the figure, comparing the presented technique with a typical GWO combined with RNN, the proposed technique acquired 98.6% accuracy rate while the typical GWO combined with RNN acquired just 94.4% accuracy rate. Both the typical GWO and the modified GWO combined with CNN acquired 86.4% and 86.3% respectively. Once again, the typical GWO and the modified GWO combined with MLP acquired 78% and 77% respectively. The other algorithms BPNN, Naïve bayes Classifier, and Random Forest have obtained 76%, 73%, and 81% accuracies respectively. The Scheme and parameters of the last three algorithms are shown in tables 4.22, 4.23, and 4.24.

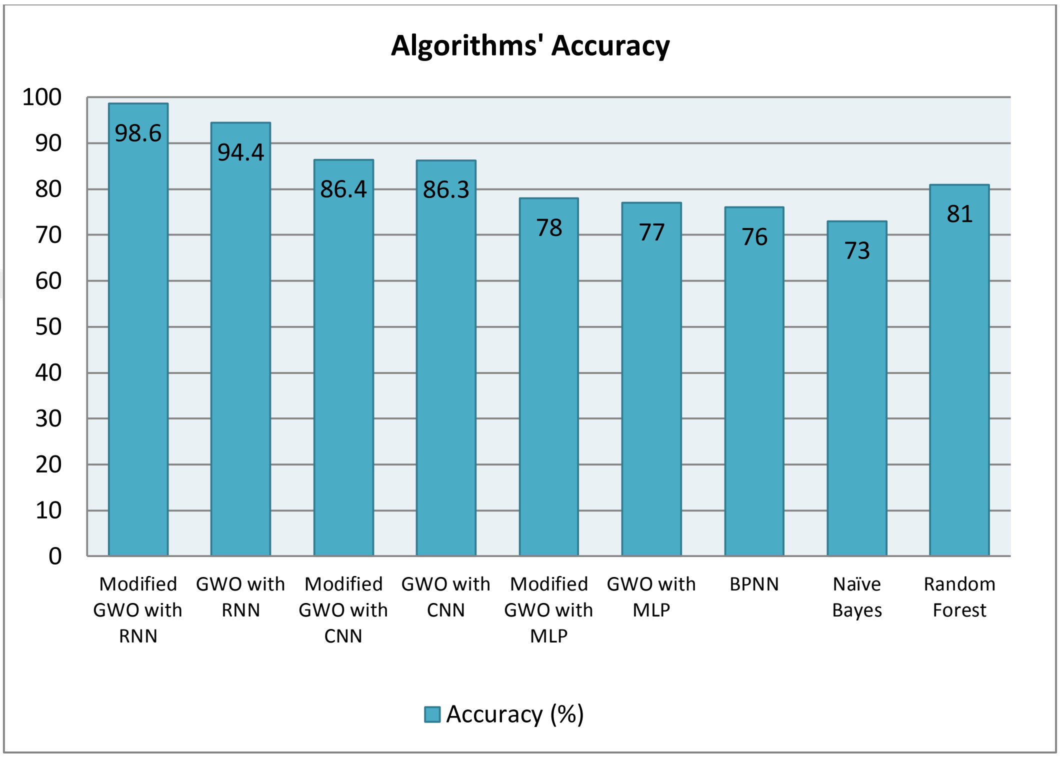

Figure 4.1 Accuracies of the research algorithms

Table 4.22 Parameters of BPNN

Test Mode	Batch Size	No. Hidden Layers	Learning Rate	Momentum	Seed	Training Time
10-fold cross-validation	100	1	0.3	0.2	0	500

Table 4.23 Parameters of Naïve bayes Classifier

Test Mode	Batch Size	Number of Decimal Places	Use Kernal Estimator	Use supervised Discretization
10-fold cross-validation	100	2	False	False

Table 4.24 Parameters of Random Forest

Test Mode	Batch Size	Max Depth	Number of Decimal Places	No. Executionn Slots	Seed	No. Iterations
10-fold cross-validation	100	0	2	1	0	100

The detailed statistics of the algorithms RNN, CNN, and MLP with both modified GWO and the standard GWO is shown in Table 4.25. The same implementation was used in these methods. Search agents number that defines number of grey wolves and iteration were 50 and 75 for all the algorithms Number of connections in the neural network algorithms RNN, CNN, and MLP which is the number of the dimension of the problem for the GWO were 511, 544, and 528 consequently. All the techniques have 1 hidden layer in the neural network except RNN with both modified and standard GWO. As we see, the highest accuracy was obtained by modified GWO with RNN.

Table 4.25 Statistics of the used techniques

Algorithm	No. Connections	Search Agents No.	Iteration(s)	No. Hidden Layers	Testing Rate
Modified GWO with RNN	511	50	75	2	98.60%
GWO with RNN	511	50	75	2	94.40%
Modified GWO with CNN	544	50	75	1	84.40%
GWO with CNN	544	50	75	1	86.35%
Modified GWO with MLP	528	50	75	1	78.00%
GWO with MLP	528	50	75	1	77.05%

If we analyze the results, we can see that RNN outperforms the other neural network types. Whenever we use the modified GWO with the used neural networks, the accuracy is greater than one which uses the standard GWO specifically with RNN. Also, the CNN outperforms MLP with both the modified and the standard GWO. In the algorithms, we used two hidden layers for the RNN and one hidden layer for the other neural network algorithms.

There is another feature that makes the RNN outperform other neural network models which is the dimension of the problem or number of connections. These connections are assigned as the positions of the wolves in the GWO. The GWO with least number of positions updates its positions faster than the one with more number of positions since it needs lesser time to update the positions. Therefore, the RNN finishes the process earlier than the other used neural network types.

5. CONCLUSIONS AND RECOMMENDATIONS

5.1. Conclusions

In this thesis, to classifying students in a course, a technique was proposed depending on the academic environments, social settings, and the students' achievements. In the presented classification approach a modified GWO to optimize the weights and biases of a RNN structure was used. Adding an extra best solution to the population of the wolves was the modulation in the GWO. Furthermore, the average distance of the best wolves was taken into consideration instead of taking separate distances of the best wolves. We used the basics of the simple kind of RNNs as a classifier based on a MLP with two hidden layers to predict students' performance.

The neural network was developed as a four-layered perceptron and was trained through the RNN with the modified GWO. The outcomes and the results of the system comparing with results obtained by other researches are highly recommended to be useful in various directions. The approach outperforms the study [47] which has used the same dataset. In the study, researchers used BPNN for training and testing. The experiments showed that the BPNN predicted 83.33% while in our research we have used a new approach which is faster and the experimental results showed that the modified GWO can give very competitive results in the sense of enhanced avoidance for local optima. Moreover, if the modified GWO is iterated with an adequate number of iterations then it converges to a solution which is more preferable than a random former solution. Additionally, the results provide a high accuracy level in classification and approximation of the suggested trainer.

Several conclusions are addressed about the techniques employed and data samples for the classification of students through taking the results of all the models that were considered in this thesis. Giving certain key points in order to identify the advantages and disadvantages of the approaches, the following points are the important conclusions:

- The performance and the accuracy of the system have been greatly impacted by the data samples. In addition, using cross validation and re-balancing the data have increased the accuracy of the system.

- Using feature selection to remove the feature that has zero influence on the output value and to eliminate one of the attributes from the features which gave similar impact on the output value made the classification more accurate.

- From the data analysis, we concluded that the features Module Test – General University Exam and department and tutors are the most related factors affecting academic performance. They have the most significant weights among the rest features.

- The features Overall Score at National Exam and Department are in the second level to be related to the output class. The other features are also important factors in the academic performance. Although they are less related to the performance of the students than the features which are mentioned in the previous point but their effect is necessary to be included in the prediction procedure and the prediction will be less accurate in case of excluding them in the model.

- Depending on the proposed technique, in the module's final exam, the students will be distributed to be passed or failed with 98.6% success rate.

- Based on the acquired results, the proposed approach which is the modified GWO combined with RNN outperformed other techniques such as a typical GWO combined with RNN, modified and typical GWO combined with CNN, modified and typical GWO combined with MLP, BPNN, Naïve bayes Classifier, and Random Forest with 98.6% accuracy rate.

- Depending on the accuracy rates of the techniques, it is found that the modification on the GWO observed great impacts as another best solution is used to update the position of the search agents.

5.2. Recommendations

The proposed technique gives satisfactory results, but improvements could still be considered to enhance the system. For the future work, we have listed some of the most important recommendations and key ideas below:

- For future, it is recommended to enhance the approach work by using different classifiers such as Support Vector Machines or presenting other natured inspired optimization algorithms to predict student’s performance and outcomes.

- Increasing size of dataset by adding more features and increasing number of samples should make the system more accurate.

- Classifying students to more than two output class labels instead of just passed and failed for example ladling as failed, accepted, medium, good, very good, and excellent.

- Using the network model for other datasets and other applications.

REFERENCES

[1] **Guo, B., Zhang, R., Xu, G., Shi, C. and Yang, L.**, 2015, July. Predicting students performance in educational data mining. In Educational Technology (ISET), 2015 International Symposium on (pp. 125-128). IEEE.

[2] **Baker, R.S. and Yacef, K.**, 2009. The state of educational data mining in 2009: A review and future visions. JEDM-Journal of Educational Data Mining, 1(1), pp.3-17.

[3] **Romero, C. and Ventura, S.**, 2010. Educational data mining: a review of the state of the art. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 40(6), pp.601-618.

[4] **Umamaheswari, K. and Niraimathi, S.**, 2013. A study on student data analysis using data mining techniques. International Journal of Advanced Research in Computer Science and Software Engineering, 3(8), pp.117-20.

[5] **Han, J., Pei, J. and Kamber, M.**, 2011. Data mining: concepts and techniques. Elsevier.

[6] **McCulloch, W.S. and Pitts, W.**, 1943. A logical calculus of the ideas immanent in nervous activity. Bulletin of mathematical biology, 5(4), pp.115-133.

[7] **Bebis, G. and Georgiopoulos, M.**, 1994. Feed-forward neural networks. IEEE Potentials, 13(4), pp.27-31.

[8] **Kohonen, T.**, 1998. The self-organizing map. Neurocomputing, 21(1), pp.1-6.

[9] **Park, J. and Sandberg, I.W.**, 1993. Approximation and radial-basis-function networks. Neural computation, 5(2), pp.305-316.

[10] **Dorffner, G.**, 1996. Neural networks for time series processing. In Neural network world.

[11] **Ghosh-Dastidar, S. and Adeli, H.**, 2009. Spiking neural networks. International journal of neural systems, 19(04), pp.295-308.

[12] **Hagan, M.T., Demuth, H.B. and Beale, M.H.**, 1996. Neural network design, PWS Pub. Co., Boston, 3632.

[13] **Sikder, M.F., Uddin, M.J. and Halder, S.**, 2016, May. Predicting students yearly performance using neural network: A case study of BSMRSTU. In Informatics, Electronics and Vision (ICIEV), 2016 5th International Conference on (pp. 524-529). IEEE.

[14] **Baradwaj, B.K. and Pal, S.**, 2012. Mining educational data to analyze students' performance. arXiv preprint arXiv:1201.3417.

[15] **Alfiani, A.P. and Wulandari, F.A.**, 2015. Mapping Student's Performance Based on Data Mining Approach (A Case Study). Agriculture and Agricultural Science Procedia, 3, pp.173-177.

[16] **Mishra, T., Kumar, D. and Gupta, S.**, 2014, February. Mining students' data for prediction performance. In Advanced Computing & Communication Technologies (ACCT), 2014 Fourth International Conference on (pp. 255-262). IEEE.

[17] **Karamouzis, S.T. and Vrettos, A.**, 2008. An artificial neural network for predicting student graduation outcomes. In Proceedings of the World Congress on Engineering and Computer Science (pp. 991-994).

[18] **Rashid, T.A. and Ahmad, H.A.**, 2016. Lecturer performance system using neural network with Particle Swarm Optimization. Computer Applications in Engineering Education, 24(4), pp.629-638.

[19] **Quadri, M.M. and Kalyankar, N.V.**, 2010. Drop out feature of student data for academic performance using decision tree techniques. Global Journal of Computer Science and Technology.

[20] **Yukselturk, E., Ozekes, S. and Türel, Y.K.**, 2014. Predicting dropout student: an application of data mining methods in an online education program. European Journal of Open, Distance and E-learning, 17(1), pp.118-133.

[21] **Parack, S., Zahid, Z. and Merchant, F.**, 2012, January. Application of data mining in educational databases for predicting academic trends and patterns. In Technology

Enhanced Education (ICTEE), 2012 IEEE International Conference on (pp. 1-4). IEEE.

[22] **Borkar, S. and Rajeswari, K.**, 2013. Predicting students academic performance using education data mining. International Journal of Computer Science and Mobile Computing (IJCSMC), 2(7), pp.273-279.

[23] **Van Dooren, W., Bouckaert, G. and Halligan, J.**, 2015. Performance management in the public sector. Routledge.

[24] **Japkowicz, N.**, 2000, July. Learning from imbalanced data sets: a comparison of various strategies. In AAAI workshop on learning from imbalanced data sets (Vol. 68, pp. 10-15).

[25] **Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P.**, 2002. SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, pp.321-357.

[26] **Amin, H.U. and Khan, A.R.**, 2009. Acquiring Knowledge for Evaluation of Teachers Performance in Higher Education Using a Questionnaire. arXiv preprint arXiv:0906.4663.

[27] **Do, Q.H. and Chen, J.F.**, 2013. A neuro-fuzzy approach in the classification of students' academic performance. Computational intelligence and neuroscience, 2013, p.6.

[28] **Chaudhari, O.K., Khot, P.G. and Deshmukh, K.C.**, 2012. Soft computing model for academic performance of teachers using fuzzy logic. British Journal of Applied Science & Technology, 2(2), p.213.

[29] **Khan, A.R., Amin, H.U. and Rehman, Z.U.**, 2011. Application of Expert System with Fuzzy Logic in Teachers 'Performance Evaluation. International Journal of Advanced Computer Science and Applications, 2(2), pp.51-57.

[30] **Abidin, A.I., Setu, I.A., Yong, S.P., Foong, O.M. and Ahmad, J.**, 2008. Classifying student academic performance: A hybrid approach. In IMECS) International MultiConference of Engineers and Computer Scientists, Hong Kong.

[31] **Huang, Z., Chen, H., Hsu, C.J., Chen, W.H. and Wu, S.**, 2004. Credit rating analysis with support vector machines and neural networks: a market comparative study. Decision support systems, 37(4), pp.543-558.

[32] **Reed, R. and Marks, R.J.**, 1999. Neural smithing: supervised learning in feedforward artificial neural networks. Mit Press.

[33] **Caruana, R. and Niculescu-Mizil, A.**, 2006, June. An empirical comparison of supervised learning algorithms. In Proceedings of the 23rd international conference on Machine learning (pp. 161-168). ACM.

[34] **Hinton, G.E. and Sejnowski, T.J. eds.**, 1999. Unsupervised learning: foundations of neural computation. MIT press.

[35] **Wang, D.**, 2001. Unsupervised learning: foundations of neural computation. AI Magazine, 22(2), p.101.

[36] **Mirjalili, S.**, 2015. How effective is the Grey Wolf optimizer in training multi-layer perceptrons. *Applied Intelligence*, *43*(1), p.150.

[37] **Mirjalili, S. and Sadiq, A.S.**, 2011, May. Magnetic optimization algorithm for training multi layer perceptron. In Communication software and networks (ICCSN), 2011 IEEE 3rd international conference on (pp. 42-46). IEEE.

[38] **Elman, J.L.**, 1990. Finding structure in time. Cognitive science, 14(2), pp.179-211.

[39] **Mesnil, G., He, X., Deng, L. and Bengio, Y.**, 2013, August. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In Interspeech (pp. 3771-3775).

[40] **Fahlman, S.E. and Lebiere, C.**, 1990. The cascade-correlation learning architecture. In Advances in neural information processing systems (pp. 524-532).

[41] **Pham, D.T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S. and Zaidi, M.**, 2011, July. The bees algorithm-A novel tool for complex optimisation. In Intelligent Production Machines and Systems-2nd I* PROMS Virtual International Conference (3-14 July 2006). sn.

[42] **Dorigo, M., Birattari, M. and Stutzle, T.**, 2006. Ant colony optimization. IEEE computational intelligence magazine, 1(4), pp.28-39.

[43] **Kennedy, J. and Eberhart, R.**, 1995, November. Particle swarm optimization. In Neural Networks, 1995. Proceedings., IEEE International Conference on(Vol. 4, pp. 1942-1948). IEEE.

[44] **Mirjalili, S., Mirjalili, S.M. and Lewis, A.**, 2014. Grey wolf optimizer. Advances in Engineering Software, 69, pp.46-61.

[45] **Fawcett, T.**, 2006. An introduction to ROC analysis. Pattern recognition letters, 27(8), pp.861-874.

[46] **Powers, D.M.**, 2011. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.

[47] **Rashid, T.A. and Aziz, N.K.**, 2016. Student Academic Performance Using Artificial Intelligence. ZANCO Journal of Pure and Applied Sciences, 28(2).

CURRICULUM VITAE

Personal Information

- Name: Dosti Kheder Abbas
- Date Of Birth : August, 1989
- Nationality: Iraqi
- Address: Erbil, Iraq
- Mobile: 009647507459583
- E-mail: dostyeng@gmail.com

Education

- (2008) High School, Shaheed Doctor Abdulrahman High School.
- (2012) BSc. In Software Engineering, Salahaddin University.
- (Sep - 2015) Enrolled MSc in Software Engineering, Firat University, Elazig, Turkey.

Language

- Mother Tongue: Kurdish language.
- English: Good spoken, written, reading and listening.
- Arabic: Moderate spoken, good written, reading and listening.

Work Experience

- IT support at Soran University since 2013.
- IT support since 2016 at Pearson VUE Test Center, Soran University.
- Teaching at Soran Private Computer Institute for 2014/2015.

Publication

- Kheder, D. A. & Türel, Y. K. (2016). An Analysis of Data Mining Techniques used in Academic Performance, Presented in 4th International Instructional Technologies & Teacher Education Symposium, 06-08 October 2016, Elazig, Turkey.

  Accessible at:

  http://www.ittes2016.org/up/belge/37/tammetin-10ocakguncelleme.pdf.